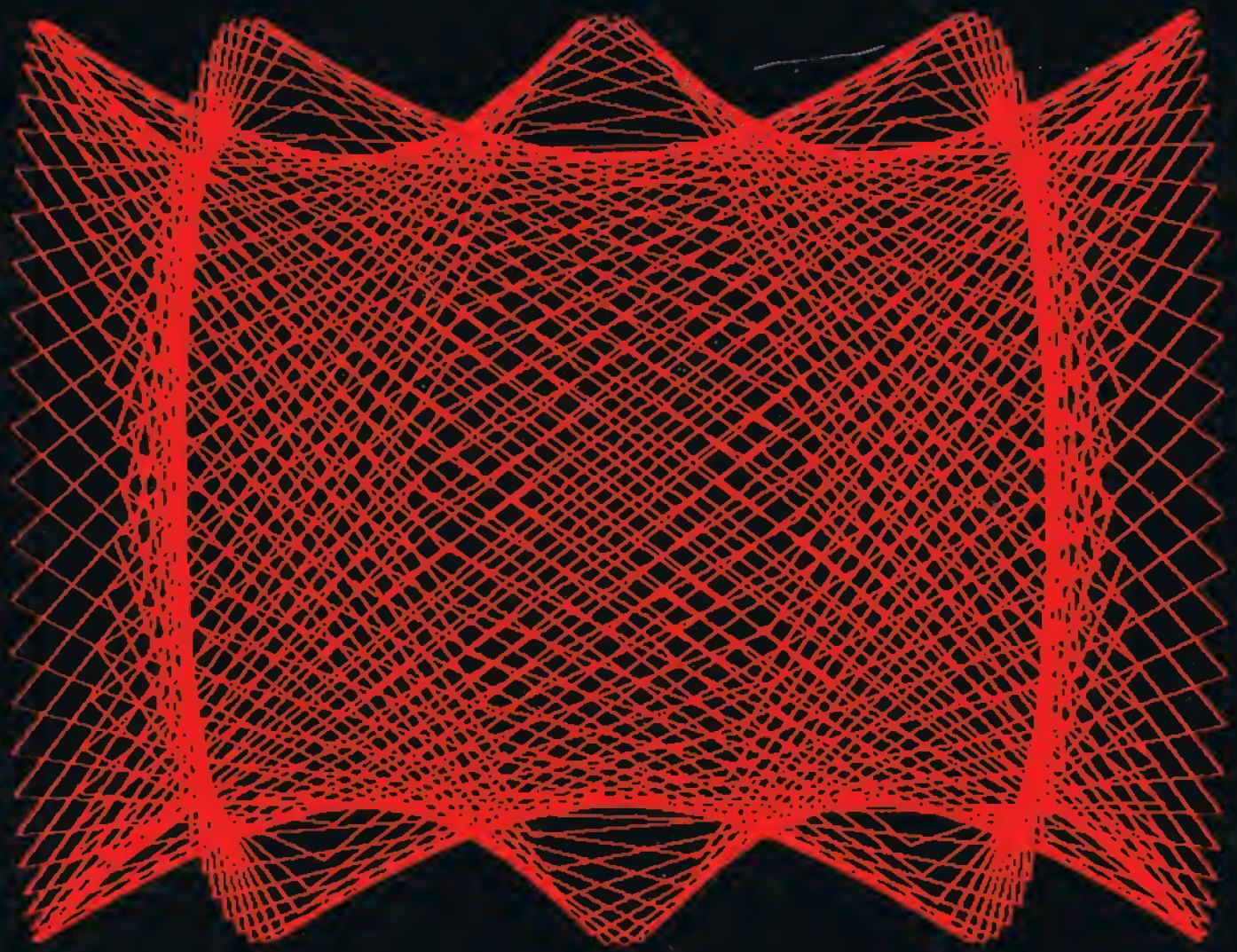
OSBORNE/McGraw-Hill

Some Common BASIC Programs

PET™/CBM™ Edition



by Lon Poole
Mary Borchers
Carroll Donahue

Some Common BASIC Programs

PET™/CBM™ Edition

Lon Poole
Mary Borchers
Carroll Donahue

OSBORNE/McGraw-Hill
Berkeley, California

DISCLAIMER OF WARRANTIES AND LIMITATION OF LIABILITIES

The authors have taken due care in preparing this book and the programs in it, including research, development, and testing to ascertain their effectiveness. The authors and the publishers make no expressed or implied warranty of any kind with regard to these programs nor the supplementary documentation in this book. In no event shall the authors or the publishers be liable for incidental or consequential damages in connection with or arising out of the furnishing, performance or use of any of these programs.

Published by
OSBORNE/McGraw-Hill
630 Bancroft Way
Berkeley, California 94710
U. S. A

For information on translations and book distributors outside of the U. S. A., please write OSBORNE/McGraw-Hill at the above address.

Computer-generated image on front cover recorded on a Model 4000 Color Graphic Camera System, manufactured by Matrix Instruments, Inc., 230 Pegasus Avenue, Northvale, New Jersey 07647.

SOME COMMON BASIC PROGRAMS — PET™/CBM™ EDITION

1 2 3 4 5 6 7 8 9 0 DLDL 8 7 6 5 4 3 2 1 0

ISBN 0-931988-40-3

Table of Contents

Table of Contents (Continued)

Introduction

This book is a collection of 76 practical programs written in the BASIC programming language specifically for the Commodore PET[1] or CBM[2] computers. The programs perform a variety of common useful tasks in the fields of finance, mathematics, statistics, science, and business.

You can use this book whether or not you know how to write programs in BASIC.

We do not teach you how to program in BASIC; there are probably hundreds of books trying to do that. But we do describe programs carefully and include user examples with the program listings. So if you are not familiar with BASIC, simply copy the program listings into your computer, then run the programs as illustrated in the examples.

Remarks are included in the listings to help BASIC programmers understand how each program works. They will also assist you in identifying parts of programs that you may be able to use in other programs you write. Remark statements precede the line(s) on which they comment. *REM statements should be omitted when you enter programs, since they are ignored by the computer and simply use up memory.*

Options are also included with some programs. An option is an alteration which changes the input or output format of the original program. Options may suggest ways in which you can further alter the programs. We have included a brief description, example, sample run and partial listing of each option. The partial listing includes those program statements which are changed when going from the original program to the optional program. Lines which must be altered, added or deleted are shaded in both listings.

All programs can be run on a PET or CBM computer with 8K or more of memory. You will need a cassette or floppy disk drive to save the programs on, but the programs themselves do not use cassettes or floppy disks to store or retrieve data.

The programs are designed to fit on a 40-column display screen, so they work equally well on 40 or 80-column PET/CBM computers. All program output appears on the display screen, except in the program "Check Writer," which uses a printer. You can modify any program to use a printer. Adding OPEN, CMD, PRINT#, and CLOSE statements to a program causes its output to go to the printer instead of the display screen. To do this, first find a place in the program after all INPUT statements but before the PRINT statements that produce the program output. At that point, add statements such as OPEN 4,4:CMD4. Any PRINT statements executed after these statements are executed will output to the printer rather than the display screen. To make output revert to the display screen, put statements such as PRINT#4:CLOSE 4,4 after the PRINT statements that produce the program output. Refer to the "Check Writer" program for an example of printer output. (Note especially lines 250 and 625.)

You can also alter the PRINT statements in any program to use more than 40 columns of an 80-column printer or display screen, but it is not necessary.

The CBM computer does not normally display graphics characters. Instead, it displays upper and lower case alphabetic characters. Some of the programs, like "Coordinate Plot," use the graphics characters. So if you use these programs on a CBM, you must enable the graphics characters by typing POKE 59468,12 before typing RUN. After the program is finished, type POKE 59468,14 to return the CBM to normal upper/lower case mode.

Throughout the program listings, the character ▯ is the "clear screen" character. When you are typing in a program listing, enter it by simultaneously pressing the SHIFT and CLR HOME keys. This character appears on the display screen as ▯ (rather than ▯).

All programs in this book have been tested, run, and listed on a Commodore PET computer system model 2001 with Commodore BASIC version 3.0. The sample runs and listings in this book were printed on a Commodore printer model 2022. The sample runs represent program output as it would appear on a display screen. In some cases it would be necessary to modify PRINT statements to achieve the same output directly on a printer.

[1] *PET is a registered trademerk of Commodore Business Machines, Inc.*

[2] *CBM is a registered tredemerk of Commodore Business Machines, Inc.*

Program Errors

If you encounter an error or program difficulty which you believe is not your fault, we would like to hear about it. Please write the authors in care of the publishers, and include the following information:

1) description of the error
2) data entered which caused the error
3) source listing of your program, from your computer (if possible)
4) any corrections you have

We appreciate your help in creating a book of tested BASIC programs that *anyone* can use.

Future Value of an Investment

This program calculates a future value of an investment when interest is a factor. You must provide the amount of the initial investment, the nominal interest rate, the number of compounding periods per year and the number of years of investment.

Assuming there are no additional deposits and no withdrawals, the future value is based on the following formula:

$$T = P(1 + i/N)^{N \cdot Y}$$

where: T = total value after Y years (future value)
P = initial investment
i = nominal interest rate
N = number of compounding period per year
Y = number of years

Examples:

Carl makes an investment of $6800.00 at 9.5%. If interest is compounded quarterly, what will be the value of Carl's investment in 10 years?

Mr. Smith purchases a piece of property for $16,050.00. The value of property is rising at an average of 7% per year. What may Mr. Smith expect his property to be worth in 5½ years?

```
FUTURE VALUE OF AN INVESTMENT

INITIAL INVESTMENT? 6800
NOMINAL INTEREST RATE? 9.5
COMPOUNDING PERIODS PER YEAR? 4
NUMBER OF YEARS? 10
FUTURE VALUE=$ 17388.64

MORE DATA? (1=YES,0=NO)? 1

INITIAL INVESTMENT? 16050
NOMINAL INTEREST RATE? 7
COMPOUNDING PERIODS PER YEAR? 1
NUMBER OF YEARS? 5.5
FUTURE VALUE=$ 23285.51

MORE DATA? (1=YES,0=NO)? 0
```

```
10 PRINT "□FUTURE VALUE OF AN INVESTMENT"
20 PRINT
29 REM - STATEMENTS 30 TO 100 REQUEST USER INPUT
30 PRINT "INITIAL INVESTMENT";
40 INPUT P
50 PRINT "NOMINAL INTEREST RATE";
60 INPUT I
70 PRINT"COMPOUNDING PERIODS PER YEAR";
80 INPUT N
```

```
90 PRINT "NUMBER OF YEARS";
100 INPUT Y
108 REM - CALCULATE INTEREST RATE PER PERIOD;
109 REM - CONVERT FROM PERCENT TO DECIMAL
110 I=I/N/100
119 REM - CALCULATE FUTURE VALUE BY FORMULA
120 T=P*(1+I)↑(N*Y)
129 REM - ROUND OFF TO NEAREST CENT, PRINT
130 PRINT "FUTURE VALUE=$";
135 PRINT INT(T*100+.5)/100
140 PRINT
149 REM - RESTART OR END PROGRAM? USER INPUT REQUIRED
150 PRINT "MORE DATA? (1=YES,0=NO)";
160 INPUT X
170 IF X=1 THEN 20
180 END
```

OPTION

This program allows you to enter a term of investment in whole years or decimal parts only. In some cases you may wish to enter the term of investment in years and months rather than just years. The program changes necessary follow the example listed below.

Example:

Herb invests $12,000.00 at 8% interest. Interest is compounded quarterly. What is the value of his investment at the end of 10 years and 7 months?

```
FUTURE VALUE OF AN INVESTMENT

INITIAL INVESTMENT? 12000
NOMINAL INTEREST RATE? 8
COMPOUNDING PERIODS PER YEAR? 4
NUMBER OF YEARS, MONTHS? 10,7
FUTURE VALUE=$ 27749.5

MORE DATA? (1=YES,0=NO)? 0
```

```
1 REM - OPTION 90-105
10 PRINT "FUTURE VALUE OF INVESTMENTS"
 .
 .
 .
80 INPUT N
90 PRINT "NUMBER OF YEARS, MONTHS"
100 INPUT Y0,M
104 REM - CALCULATE YEARS FROM YEARS AND MONTHS
105 Y=(12*Y0+M)/12
108 REM - CALCULATE INTEREST RATE PER PERIOD;
 .
 .
 .

180 END
```

Future Value of Regular Deposits (Annuity)

This program calculates a future value when deposits are made regularly. All deposits are equal. You must provide the amount of each deposit, the number of deposits per year, the number of years, and the nominal interest rate.

Assuming that interest is compounded with each deposit, the calculation is based on the following formula:

$$T = R \cdot \left(\frac{(1 + i/N)^{N \cdot Y} - 1}{i/N} \right)$$

where: T = total value after Y years (future value)
R = amount of regular deposits
N = number of deposits per year
Y = number of years
i = nominal interest rate

Examples:

$50.00 is transferred each month from Matt's checking account to a Christmas Club savings account with 5% interest. How much will Matt receive at the end of the year?

Tim makes annuity payments of $175.00. The interest is 5.5%. What amount will Tim have accumulated in 15 years?

```
FUTURE VALUE OF REGULAR DEPOSITS

AMOUNT OF REGULAR DEPOSITS? 50
NOMINAL INTEREST RATE? 5
# OF DEPOSITS PER YEAR? 12
NUMBER OF YEARS? 1
FUTURE VALUE = $ 613.94

MORE DATA? (1=YES, 0=NO)? 1

AMOUNT OF REGULAR DEPOSITS? 175
NOMINAL INTEREST RATE? 5.5
# OF DEPOSITS PER YEAR? 1
NUMBER OF YEARS? 15
FUTURE VALUE = $ 3921.52

MORE DATA? (1=YES, 0=NO)? 0
```

```
10 PRINT "FUTURE VALUE OF REGULAR DEPOSITS"
20 PRINT
29 REM - STATEMENTS 30 TO 100 REQUEST USER INPUT
30 PRINT "AMOUNT OF REGULAR DEPOSITS";
40 INPUT R
50 PRINT "NOMINAL INTEREST RATE";
60 INPUT I
70 PRINT "# OF DEPOSITS PER YEAR";
```

```
80 INPUT N
90 PRINT "NUMBER OF YEARS";
100 INPUT Y
108 REM - CALCULATE INTEREST RATE PER DEPOSIT.
109 REM - CONVERT FROM PERCENT TO DECIMAL
110 I=I/N/100
119 REM - CALCULATE FUTURE VALUE BY FORMULA
120 T=R*((1+I)↑(N*Y)-1)/I
129 REM - ROUND OFF TO NEAREST CENT, PRINT
130 PRINT "FUTURE VALUE = $";
135 PRINT INT(T*100+.5)/100
140 PRINT
149 REM - RESTART OR END PROGRAM? USER INPUT REQUIRED
150 PRINT "MORE DATA? (1=YES, 0=NO)";
160 INPUT X
170 IF X=1 THEN 20
180 END
```

OPTION

You may wish to enter the term of investment in years and months rather than years. The program changes necessary are listed following the example below.

Example:

How much will Ron receive in 10 years and 5 months if he transfers $50.00 each month into a trust fund with 5% interest?

```
FUTURE VALUE OF REGULAR DEPOSITS

AMOUNT OF REGULAR DEPOSITS? 50
NOMINAL INTEREST RATE? 5
# OF DEPOSITS PER YEAR? 12
NUMBER OF YEARS, MONTHS? 10,5
FUTURE VALUE = $ 8179.31

MORE DATA? (1=YES, 0=NO)? 0
```

```
1 REM - OPTION 90-105
10 PRINT "FUTURE VALUE OF REGULAR DEPOSITS"
 .
 .
 .
80 INPUT N
90 PRINT "NUMBER OF YEARS, MONTHS"
100 INPUT Y0,M
104 REM - CALCULATE YEARS FROM YEARS AND MONTHS
105 Y=(12*Y0+M)/12
108 REM - CALCULATE INTEREST RATE PER DEPOSIT
 .
 .
 .

180 END
```

Regular Deposits

This program calculates the amount required as a regular deposit to provide a stated future value in a specified time period. All deposits are equal. It is necessary for you to supply the future value, the nominal interest rate, the number of deposits per year and the number of years.

The calculation for regular deposits is based on the following formula:

$$R = T\left(\frac{i/N}{(1+i/N)^{N\cdot Y}-1}\right)$$

where: R = amount of regular deposit
T = future value
i = nominal interest rate
N = number of deposits per year
Y = number of years

Example:

Mary would like $1000.00 at the end of one year in a savings account. How much must she deposit each month at 8% interest to achieve this?

```
REGULAR DEPOSITS

TOTAL VALUE AFTER Y YEARS? 1000
NOMINAL INTEREST RATE? 8
# OF DEPOSITS PER YEAR? 12
NUMBER OF YEARS? 1
REGULAR DEPOSITS = $ 80.32

MORE DATA? (1=YES, 0=NO)? 0
```

```
10 PRINT "REGULAR DEPOSITS"
20 PRINT
29 REM - STATEMENTS 30 TO 100 REQUEST USER INPUT
30 PRINT "TOTAL VALUE AFTER Y YEARS";
40 INPUT T
50 PRINT "NOMINAL INTEREST RATE";
60 INPUT I
70 PRINT "# OF DEPOSITS PER YEAR";
80 INPUT N
90 PRINT "NUMBER OF YEARS";
100 INPUT Y
108 REM - CALCULATE INTEREST RATE PER DEPOSIT;
109 REM - CONVERT FROM PERCENT TO DECIMAL
110 I=I/N/100
119 REM - CALCULATE AMOUNT OF REGULAR DEPOSIT BY FORMULA
120 R=T*I/((I+1)↑(N*Y)-1)
129 REM - ROUND OFF TO NEAREST CENT, PRINT
```

```
130 PRINT "REGULAR DEPOSITS = $";
135 PRINT INT(R*100+.5)/100
140 PRINT
149 REM - RESTART OR END PROGRAM? USER INPUT REQUIRED
150 PRINT "MORE DATA? (1=YES, 0=NO)";
160 INPUT X
170 IF X=1 THEN 20
180 END
```

OPTION

You may wish to enter the term of investment in years and months rather than years. The program changes necessary are listed following the example below.

Example:

Ed would like to save $2000.00 for a new motorcycle. He would like to achieve this amount in 1 year and 5 months. How much must he deposit each month if his interest is 8%?

```
REGULAR DEPOSITS

TOTAL VALUE AFTER Y YEARS? 2000
NOMINAL INTEREST RATE? 8
# OF DEPOSITS PER YEAR? 12
NUMBER OF YEARS, MONTHS? 1,5
REGULAR DEPOSITS = $ 111.5

MORE DATA? (1=YES, 0=NO)? 0
```

```
1 REM - OPTION 90-105
10 PRINT "□REGULAR DEPOSITS"
 .
 .
 .
80 INPUT N
90 PRINT "NUMBER OF YEARS, MONTHS"
100 INPUT Y0,M
104 REM - CALCULATE YEARS FROM YEARS AND MONTHS
105 Y=(12*Y0+M)/12
108 REM - CALCULATE INTEREST RATE PER DEPOSIT,
 .
 .
 .

180 END
```

Regular Withdrawals from an Investment

This program calculates the maximum amount which may be withdrawn regularly from an investment over a specified time period. All withdrawals are assumed to be equal. You must provide the amount of the initial investment, the nominal interest rate, the number of withdrawals per year and the number of years.

The maximum amount of withdrawals is calculated by the following formula:

$$R = P\left(\frac{i/N}{(1+i/N)^{N\cdot Y}-1} + \frac{i}{N}\right)$$

where: R = amount of regular withdrawal
P = initial investment
i = nominal interest rate
N = number of withdrawals per year
Y = number of years

Because this program calculates a maximum amount, a balance of $0.00 will be left in your account at the end of the time period. You may withdraw any lesser amount under the same specifications and leave a remaining balance in your account.

Example:

David invests $8000.00 at 9.5%. He plans to make regular withdrawals every month for ten years, leaving nothing at the end. How much should he withdraw each time?

```
REGULAR WITHDRAWALS FROM AN INVESTMENT

INITIAL INVESTMENT? 8000
NOMINAL INTEREST RATE? 9.5
NUMBER OF WITHDRAWALS ? 12
NUMBER OF YEARS? 10
AMOUNT OF WITHDRAWALS = $ 103.52

MORE DATA? (1=YES, 0=NO)? 0
```

```
10 PRINT "□REGULAR WITHDRAWALS FROM AN INVESTMENT"
20 PRINT
29 REM - STATEMENTS 30 TO 100 REQUEST USER INPUT
30 PRINT "INITIAL INVESTMENT";
40 INPUT P
50 PRINT "NOMINAL INTEREST RATE";
60 INPUT I
70 PRINT "NUMBER OF WITHDRAWALS";
71 PRINT "PER YEAR";
80 INPUT N
90 PRINT "NUMBER OF YEARS";
100 INPUT Y
108 REM - CALCULATE INTEREST RATE PER WITHDRAWAL;
109 REM - CONVERT FROM PERCENT TO DECIMAL
```

```
110 I=I/N/100
119 REM - CALCULATE REGULAR WITHDRAWAL BY FORMULA
120 R=P*(I/((1+I)↑(N*Y)-1)+I)
125 K=INT(R*100+.5)/100
129 REM - ROUND OFF TO NEAREST CENT, PRINT
130 PRINT "AMOUNT OF WITHDRAWALS = $";K
140 PRINT
149 REM - RESTART OR END PROGRAM? USER INPUT REQUIRED
150 PRINT "MORE DATA? (1=YES, 0=NO)";
160 INPUT X
170 IF X=1 THEN 20
180 END
```

OPTION

It may be more convenient to enter the period of investment in years and months rather than just years. The program changes necessary are listed following the example below.

Example:

How much could be withdrawn each week if you have an investment of $8000.00 at 9.5% interest to be withdrawn from for 10 years and 5 months?

```
REGULAR WITHDRAWALS FROM AN INVESTMENT

INITIAL INVESTMENT? 8000
NOMINAL INTEREST RATE? 9.5
NUMBER OF WITHDRAWALS ? 52
NUMBER OF YEARS, MONTHS? 10,5
AMOUNT OF WITHDRAWALS = $ 23.28

MORE DATA? (1=YES, 0=NO)? 0
```

```
1 REM - OPTION 90-105
10 PRINT "□REGULAR WITHDRAWALS FROM AN INVESTMENT"
 .
 .
 .
80 INPUT N
90 PRINT "NUMBER OF YEARS, MONTHS"
100 INPUT Y0,M
104 REM - CALCULATE YEARS FROM YEARS AND MONTHS
105 Y=(12*Y0+M)/12
108 REM - CALCULATE INTEREST RATE PER WITHDRAWAL;
 .
 .
 .

180 END
```

Initial Investment

This program calculates the investment necessary to provide a stated future value in a specified time period. You must enter the future value of the investment, the number of years of investment, the number of compounding periods per year and the nominal interest rate.

The formula used to calculate the initial investment is as follows:

$$P = \frac{T}{(1+i/N)^{N \cdot Y}}$$

where: P = initial investment
T = future value
N = number of compounding periods per year
Y = number of years
i = nominal interest rate

Examples:

How much must you invest at 8.5% to produce $10,000.00 at the end of 10 years if interest is compounded quarterly?

Merchant Savings wishes to sell a bond which will be worth $5000.00 five years from the purchase date. Interest will be 7.9% compounded daily. How much must the bank charge for the bond?

```
INITIAL INVESTMENT

TOTAL VALUE AFTER Y YEARS? 10000
# OF COMPOUNDING PERIODS PER YEAR? 4
NUMBER OF YEARS? 10
NOMINAL INTEREST RATE? 8.5
INITIAL INVESTMENT = $ 4312.38

MORE DATA? (1=YES,0=NO)? 1

TOTAL VALUE AFTER Y YEARS? 5000
# OF COMPOUNDING PERIODS PER YEAR? 365
NUMBER OF YEARS? 5
NOMINAL INTEREST RATE? 7.9
INITIAL INVESTMENT = $ 3368.54

MORE DATA? (1=YES,0=NO)? 0
```

```
10 PRINT "INITIAL INVESTMENT"
20 PRINT
29 REM - STATEMENTS 30 TO 100 REQUEST USER INPUT
30 PRINT "TOTAL VALUE AFTER Y YEARS";
40 INPUT T
50 PRINT "# OF COMPOUNDING PERIODS PER YEAR";
60 INPUT N
```

```
70 PRINT "NUMBER OF YEARS";
80 INPUT Y
90 PRINT "NOMINAL INTEREST RATE";
100 INPUT I
108 REM - CALCULATE INTEREST RATE PER PERIOD
109 REM - CONVERT FROM PERCENT TO DECIMAL
110 I=I/N/100
119 REM - CALCULATE  INITIAL INVESTMENT BY FORMULA
120 P=T/(1+I)↑(N*Y)
129 REM - ROUND OFF TO NEAREST CENT, PRINT
130 PRINT "INITIAL INVESTMENT = $";
135 PRINT INT(P*100+.5)/100
140 PRINT
149 REM - RESTART OR END PROGRAM? USER INPUT REQUIRED
150 PRINT "MORE DATA?(1=YES,0=NO)";
160 INPUT X
170 IF X=1 THEN 20
180 END
```

OPTION

The program above allows you to enter a period of investment of whole years and decimal parts only. You may wish to enter the period of investment in years and months rather than just years. The program changes necessary are listed following the example below.

Example:

Mary wishes to invest a sum in a savings bank. In 3 years and 8 months she would like to have $4000.00 in her account. If 8% interest is compounded monthly, what amount must Mary invest?

```
INITIAL INVESTMENT

TOTAL VALUE AFTER Y YEARS? 4000
# OF COMPOUNDING PERIODS PER YEAR? 12
NUMBER OF YEARS, MONTHS? 3,8
NOMINAL INTEREST RATE? 8
INITIAL INVESTMENT = $ 2986

MORE DATA? (1=YES,0=NO)? 0
```

```
1 REM - OPTION 70-85
10 PRINT "□INITIAL INVESTMENT"
 .
 .
 .
60 INPUT N
70 PRINT "NUMBER OF YEARS, MONTHS"
80 INPUT Y0,M
84 REM - CALCULATE YEARS FROM YEARS AND MONTHS
85 Y=(12*Y0+M)/12
90 PRINT "NOMINAL INTEREST RATE";
 .
 .
 .
180 END
```

Minimum Investment for Withdrawals

This program calculates the minimum investment required to allow regular withdrawals over a specified time period. The amount calculated is dependent upon the amount of each withdrawal, the number of withdrawals per year, the number of years, and the nominal interest rate on the investment. All withdrawals are equal.

Only the least amount necessary for your investment is calculated; the program assumes a balance of $0.00 to be left at the end of the time period. Any investment larger than the amount calculated will also enable you to withdraw the desired amount, but leave a remaining balance.

Assuming that interest is compounded with each withdrawal, the calculation is based on the following formula:

$$P = \frac{R \cdot N}{i} \left(1 - \frac{1}{(1 + i/N)^{N \cdot Y}} \right)$$

where: P = initial investment
R = amount of regular withdrawal
i = nominal interest rate
N = number of withdrawals per year
Y = number of years

Example:

How much must you invest at 6% interest to allow monthly withdrawals of $100.00 for 5 years?

```
MINIMUM INVESTMENT FOR WITHDRAWALS

AMOUNT OF WITHDRAWALS? 100
NOMINAL INTEREST RATE? 6
# OF WITHDRAWALS PER YEAR? 12
NUMBER OF YEARS? 5
MINIMUM INVESTMENT = $ 5172.56

MORE DATA? (1=YES, 0=NO)? 0
```

```
10 PRINT "MINIMUM INVESTMENT ";
11 PRINT "FOR WITHDRAWALS"
20 PRINT
29 REM - STATEMENTS 30 TO 100 REQUEST USER INPUT
30 PRINT "AMOUNT OF WITHDRAWALS";
40 INPUT R
50 PRINT "NOMINAL INTEREST RATE";
60 INPUT I
70 PRINT "# OF WITHDRAWALS PER YEAR";
80 INPUT N
90 PRINT "NUMBER OF YEARS";
100 INPUT Y
109 REM - CONVERT FROM PERCENT TO DECIMAL
```

```
110 I=I/100
119 REM - CALCULATE MINIMUM INVESTMENT BY FORMULA
120 P=R*N/I*(1-1/((1+I/N)↑(N*Y)))
129 REM - ROUND OFF TO NEAREST CENT, PRINT
130 PRINT "MINIMUM INVESTMENT = $";
135 PRINT INT(100*P+.5)/100
140 PRINT
149 REM - RESTART OR END PROGRAM? USER INPUT REQUIRED
150 PRINT "MORE DATA? (1=YES, 0=NO)";
160 INPUT X
170 IF X=1 THEN 20
180 END
```

OPTION

It may be more convenient to enter the term of investment in years and months rather than years. The program changes necessary are listed following the example below.

Example:

Tony withdrew $250.00 monthly for 6 years and 5 months. How much was his initial investment at 6% interest?

```
MINIMUM INVESTMENT FOR WITHDRAWALS

AMOUNT OF WITHDRAWALS? 250
NOMINAL INTEREST RATE? 6
# OF WITHDRAWALS PER YEAR? 12
NUMBER OF YEARS, MONTHS? 6,5
MINIMUM INVESTMENT = $ 15944.81

MORE DATA? (1=YES, 0=NO)? 0
```

```
1 REM - OPTION 90-105
10 PRINT "MINIMUM INVESTMENT
11 PRINT "FOR WITHDRAWALS"
 .
 .
 .
80 INPUT N
90 PRINT "NUMBER OF YEARS, MONTHS"
100 INPUT Y0,M
104 REM - CALCULATE YEARS FROM YEARS AND MONTHS
105 Y=(12*Y0+M)/12
109 REM - CONVERT FROM PERCENT TO DECIMAL
 .
 .
 .

180 END
```

Nominal Interest Rate on Investments

This program calculates the nominal interest rate for a known initial investment which amounts to a known future value in a specified period of time. The nominal interest rate is usually subdivided for compounding purposes.

"Nominal Interest Rate" is based on the following formula:

$$i = N(T/P)^{\frac{1}{N \cdot Y}} - N$$

where:
- i = nominal interest rate
- P = initial investment
- T = future value
- N = number of compounding periods per year
- Y = number of years

The nominal interest rate is expressed as a yearly rate even though the interest rate used when compounding interest is $\frac{i}{N}$. The nominal interest rate will be less than the effective interest rate when interest is compounded more than once a year. This is because the nominal rate stated does not take into account interest compounded on interest earned in earlier periods of each year. For example, the schedule of earned interest on $100.00 at 5% compounded quarterly would be:

PERIOD	BALANCE	$\frac{i/100}{N}$		INTEREST	NEW BALANCE
1	$100.00	• .0125	=	$1.25	$101.25
2	$101.25	• .0125	=	$1.27	$102.52
3	$102.52	• .0125	=	$1.28	$103.80
4	$103.80	• .0125	=	$1.30	$105.10

The *effective* interest rate in the example is 5.1%, although the *nominal* rate is 5%.

Examples:

Jane invests $945.00 in a savings bank. Four and a half years later her investment amounts to $1309.79. If interest is compounded monthly, what is the nominal interest rate offered by the bank?

Dick invests $3,000.00. Ten years later he has earned $1,576.00 in interest. If interest is compounded each month, what is the nominal interest rate on the account?

```
NOMINAL INTEREST RATE ON INVESTMENTS

PRINCIPAL? 945
TOTAL VALUE? 1309.79
NUMBER OF YEARS? 4.5
# OF COMPOUNDING PERIODS PER YEAR? 12
NOMINAL INTEREST RATE = 7.27613009 %

MORE DATA? (1=YES, 0=NO)? 1
```

```
PRINCIPAL? 3000
TOTAL VALUE? 4576
NUMBER OF YEARS? 10
# OF COMPOUNDING PERIODS PER YEAR? 12
NOMINAL INTEREST RATE = 4.2295659 %

MORE DATA? (1=YES, 0=NO)? 0
```

```
10 PRINT "NOMINAL INTEREST RATE ";
11 PRINT "ON INVESTMENTS"
20 PRINT
29 REM - STATEMENTS 30 TO 100 REQUEST USER INPUT
30 PRINT "PRINCIPAL";
40 INPUT P
50 PRINT "TOTAL VALUE";
60 INPUT T
70 PRINT "NUMBER OF YEARS";
80 INPUT Y
90 PRINT "# OF COMPOUNDING PERIODS ";
95 PRINT "PER YEAR";
100 INPUT N
109 REM - CALCULATE NOMINAL INTEREST RATE BY FORMULA, PRINT
110 I2=N*((T/P)↑(1/(N*Y))-1)*100
120 PRINT "NOMINAL INTEREST RATE =";
121 PRINT I2;"%"
130 PRINT
139 REM - RESTART OR END PROGRAM? USER INPUT REQUIRED
140 PRINT "MORE DATA? (1=YES, 0=NO)";
150 INPUT X
160 IF X=1 THEN 20
170 END
```

Effective Interest Rate on Investments

This program calculates the effective interest rate for a known initial investment which amounts to a known future value in a specified period of time. This rate expresses the actual rate of interest earned annually on the investment.

The effective interest rate is calculated by the following formula:

$$\text{effective interest rate} = \left(\frac{\text{future value}}{\text{initial investment}}\right)^{1/\text{years}} - 1$$

You may calculate the effective interest rate on amounts you have already invested and accrued interest. Or you may calculate the effective interest rate necessary to enable a principal to reach a hypothetical value in a specified amount of time. For instance, if you invest $5000.00 in a bank and desire $6800.00 after six years, you will predict the effective interest rate the bank must pay in order to achieve this.

"Effective Interest Rate" may also be used to calculate the effective percent of depreciation of an investment. Take your car, for example. If you bought it for $7534.00 and sold it for $3555.00 three years later, you will find that its actual depreciation (a negative interest rate) was approximately 22% each year.

Examples:

Jane deposits $945.00 in a savings bank. Four and a half years later her account has $1309.79. What actual percent of her initial investment did the bank pay annually?

Dick bought his car in 1970 for $7534.84 and sold it in 1973 for $3555.00. What was its effective rate of depreciation?

```
EFFECTIVE INTEREST RATE ON INVESTMENTS

INITIAL INVESTMENT? 945
TOTAL VALUE AFTER Y YEARS? 1309.79
NUMBER OF YEARS? 4.5
ANNUAL INTEREST RATE = 7.52375279 %

MORE DATA? (1=YES,0=NO)? 1

INITIAL INVESTMENT? 7534.84
TOTAL VALUE AFTER Y YEARS? 3555
NUMBER OF YEARS? 3
ANNUAL INTEREST RATE = -22.1506143 %

MORE DATA? (1=YES,0=NO)? 0
```

```
10 PRINT "EFFECTIVE INTEREST RATE ";
11 PRINT "ON INVESTMENTS"
20 PRINT
29 REM - STATEMENTS 30 TO 80 REQUEST USER INPUT
30 PRINT "INITIAL INVESTMENT";
40 INPUT P
```

```
50 PRINT "TOTAL VALUE AFTER Y YEARS";
60 INPUT T
70 PRINT "NUMBER OF YEARS";
80 INPUT Y
89 REM - CALCULATE EFFECTIVE INTEREST RATE, PRINT
90 PRINT "ANNUAL INTEREST RATE =";
99 REM - PRINT BLANK LINE TO SEPARATE DATA FROM QUESTION
100 PRINT ((T/P)↑(1/Y)-1)*100;"%"
109 PRINT
110 REM - RESTART OR END PROGRAM? USER INPUT REQUIRED
120 PRINT "MORE DATA? (1=YES,0=NO)";
130 INPUT X
140 IF X=1 THEN 20
150 END
```

Earned Interest Table

This program calculates and prints an earned interest table for investments. The schedule contains the following outputs:

1) Periodic balance
2) Interest accumulated between two periods
3) Total interest accumulated
4) Effective interest rate

These outputs may be calculated for a single investment or for an initial investment with regular deposits or withdrawals. If the table is to be tabulated for a single investment, you must provide the amount of the initial investment, the nominal interest rate, and the number of compounding periods per year. Your new balance will be printed a maximum of four times per year. If interest is compounded less than four times per year, your new balance will be posted with each interest computation.

If the table is tabulated for regular deposits or withdrawals, you must provide the amount of the initial investment, the nominal interest rate, the number of deposits or withdrawals per year and their amount. In this case it is assumed interest is compounded daily (360-day year). Your new balance will be printed at each deposit or withdrawal.

Examples:

Sally invests $2000.00 at 9.5% in a trust fund for ten years. Interest is compounded monthly. What is her yearly balance and earned interest for the last two years?

John deposits $1000.00 at 8% in a passbook savings account. From each monthly paycheck $50.00 is deposited in this account. What is the earned interest table for the first year of this account?

Ted deposits $1000.00 at 8% in his savings. Each quarter he withdraws $150.00. What is the earned interest table for year one of this account?

```
EARNED INTEREST TABLE

PRINCIPAL? 2000
NOMINAL INTEREST RATE? 9.5
# OF DEPOSITS/WITHDRAWALS PER YEAR? 0
# OF COMPOUNDING PERIODS PER YEAR? 12
START WITH WHAT YEAR? 9
END PRINTING WITH WHAT YEAR? 10
```

```
          EARNED INTEREST TABLE

PRINCIPAL $ 2000    AT  9.5 % NOMINAL
              FOR 10 YEARS

EFFECTIVE INTEREST RATE 9.92 % PER YEAR

YEAR    BALANCE     INTEREST     ACCUM.INT.

 9      4365.87        2365.86    2365.87
        4470.38        104.51     2470.38
        4577.39        107.01     2577.39
        4686.97        109.58     2686.97

 10     4799.17        112.2      2799.17
        4914.06        114.89     2914.06
        5031.7         117.64     3031.7
        5152.15        120.45     3152.15

 11
CHANGE DATA AND RECOMPUTE?
(1=YES,0=NO)? 1

EARNED INTEREST TABLE

PRINCIPAL? 1000
NOMINAL INTEREST RATE? 8
# OF DEPOSITS/WITHDRAWALS PER YEAR? 12
AMOUNT OF DEPOSIT/WITHDRAWAL? 50
START WITH WHAT YEAR? 1
END PRINTING WITH WHAT YEAR? 1


          EARNED INTEREST TABLE

PRINCIPAL $ 1000    AT  8 % NOMINAL
              FOR 1 YEARS

REGULAR DEPOSITS/WITHDRAWALS $ 50
            12 TIMES PER YEAR

EFFECTIVE INTEREST RATE 8.33 % PER YEAR

YEAR    BALANCE     INTEREST     ACCUM.INT.

 1      1056.7         6.7        6.7
        1113.78        7.08       13.78
        1171.24        7.46       21.24
        1229.08        7.84       29.08
        1287.32        8.23       37.32
        1345.94        8.62       45.94
        1404.95        9.01       54.95
        1464.36        9.41       64.36
        1524.17        9.81       74.17
        1584.17        10.21      84.38
        1644.98        10.61      94.98
        1706           11.01      106
```

```
ENTER 'C' TO CONTINUE? C


CHANGE DATA AND RECOMPUTE?
(1=YES,0=NO)? 1


EARNED INTEREST TABLE

PRINCIPAL? 1000
NOMINAL INTEREST RATE? 8
# OF DEPOSITS/WITHDRAWALS PER YEAR? 4
AMOUNT OF DEPOSIT/WITHDRAWAL? -150
START WITH WHAT YEAR? 1
END PRINTING WITH WHAT YEAR? 1


          EARNED INTEREST TABLE

PRINCIPAL $ 1000   AT  8 % NOMINAL
              FOR 1 YEARS

REGULAR DEPOSITS/WITHDRAWALS $-150
             4 TIMES PER YEAR

EFFECTIVE INTEREST RATE 8.33 % PER YEAR

YEAR    BALANCE     INTEREST     ACCUM.INT.

 1      870.17         20.17      20.17
        737.71         17.54      37.71
        602.58         14.87      52.58
        464.72         12.14      64.72

CHANGE DATA AND RECOMPUTE?
(1=YES,0=NO)? 0
```

```
10 PRINT "EARNED INTEREST TABLE"
20 PRINT
24 REM - ROUND OFF FUNCTION
25 DEFFNR(X)=INT(X*100+.5)/100
29 REM - STATEMENTS 30 TO 230 REQUEST USER INPUT
30 PRINT "PRINCIPAL";
40 INPUT P
50 PRINT "NOMINAL INTEREST RATE";
60 INPUT I
69 REM - CONVERT PERCENT TO DECIMAL
70 I=I/100
80 PRINT "# OF DEPOSITS/WITHDRAWALS PER YEAR";
90 INPUT N1
99 REM - DON'T ASK FOR AMOUNT IF FREQUENCY IS ZERO
100 IF N1=0 THEN 160
108 REM - DEPOSITS ARE ENTERED AS A POSITIVE NUMBER
109 REM - WITHDRAWALS ARE ENTERED AS A NEGATIVE NUMBER
110 PRINT "AMOUNT OF DEPOSIT/WITHDRAWAL";
120 INPUT R
```

```
129 REM - INTEREST IS COMPOUNDED DAILY
130 N=360
139 REM - PRINT AT EACH DEPOSIT/WITHDRAWAL
140 L2=N1
150 GOTO 200
160 PRINT "# OF COMPOUNDING PERIODS PER YEAR";
170 INPUT N
180 N1=0
189 REM - PRINT FOUR TIMES EACH YEAR
190 L2=4
200 PRINT "START WITH WHAT YEAR";
210 INPUT X
220 PRINT "END PRINTING WITH WHAT YEAR";
230 INPUT Y
239 REM - START PRINTING AT THE BEGINNING OF A YEAR
240 X=INT(X)
249 REM - INITIATE RUNNING TOTALS
250 B0=P
260 I1=0
270 I2=0
280 I2=0
290 K=66
300 P1=4
310 FOR J0=1 TO INT(Y)+1
313 REM - START PRINTING?
314 REM - IF FIRST YEAR, SKIP CHECK FOR FULL SCREEN
315 IF J0=X THEN 370
320 IF J0<X THEN 480
330 IF K<22 THEN 470
338 REM - FULL SCREEN (22 LINES)?
339 REM - IF YES, CLEAR SCREEN, PRINT HEADINGS
342 REM - WAIT FOR OPERATOR CUE TO GO TO NEXT SCREEN
344 PRINT"ENTER 'C' TO CONTINUE";
345 INPUT W$
370 K=6+L2
380 PRINT "□          EARNED INTEREST TABLE":PRINT
390 PRINT "PRINCIPAL $";P;"  AT ";I*100;"% NOMINAL"
395 PRINT"               FOR";Y;"YEARS":PRINT
399 REM - SKIP DEPOSIT/WITHDRAWAL HEADING IF THERE ARE NONE
400 IF N1=0 THEN 430
410 PRINT "REGULAR DEPOSITS/WITHDRAWALS $";R
415 PRINT"              ";N1;"TIMES PER YEAR":PRINT
419 REM - K COUNTS THE NUMBER OF PRINTED LINES PER PAGE
430 PRINT "EFFECTIVE INTEREST RATE";
435 PRINT FNR(100*((1+I/N)↑N-1));
436 PRINT "% PER YEAR"
440 PRINT
450 PRINT "YEAR   BALANCE     INTEREST     ACCUM.INT."
459 REM - CALCULATE INTEREST
460 PRINT
469 REM - PRINT YEAR NUMBER
470 PRINT J0;
480 L1=1
490 N2=1
500 P2=1
510 FOR J1=1 TO N
519 REM - DEPOSIT/WITHDRAW ANY MORE THIS YEAR?
```

```
520 IF N2>N1 THEN 560
529 REM - TIME TO MAKE DEPOSIT/WITHDRAWAL?
530 IF N2/N1>J1/N THEN 560
539 REM - CALCULATE NEW BALANCE
540 B0=B0+R
549 REM - COUNT DEPOSITS/WITHDRAWALS MADE PER YEAR
550 N2=N2+1
560 B2=B0*(1+I/N)
569 REM - I1=AMOUNT INTEREST WITH EACH COMPOUNDING PERIOD
570 I1=B2-B0
579 REM - I3=AMOUNT INTEREST ACCUMULATED BETWEEN POSTINGS
580 I3=I3+I1
589 REM - I2=TOTAL INTEREST ACCUMULATED TO DATE
590 I2=I2+I1
599 REM - ROUND AT INTEREST POSTING TIME
600 IF P2/P1>J1/N THEN 640
610 I2=FNR(I2)
620 B2=FNR(B2)
630 P2=P2+1
639 REM - YEAR TO START PRINTING?
640 IF J0<X THEN 710
649 REM - TIME TO PRINT A LINE?
650 IF J1/N<L1/L2 THEN 710
660 L1=L1+1
670 PRINT TAB(6);FNR(B2),FNR(I3),FNR(I2)
679 REM - INTEREST POSTED, REINITIALIZE INTEREST ACCUMULATED BETWEEN
    POSTINGS
680 I3=0
710 B0=B2
719 REM - NO MORE LINES TO PRINT IN LAST YEAR?
720 IF J0+J1/N-1>Y THEN 780
730 NEXT J1
739 REM - START PRINTING?
740 IF J0<X THEN 770
750 PRINT
760 K=K+1+L2
770 NEXT J0
780 PRINT
789 REM - RESTART OR END PROGRAM? USER INPUT REQUIRED
790 PRINT "CHANGE DATA AND RECOMPUTE?"
795 PRINT "(1=YES,0=NO)";
800 INPUT Z
810 PRINT
820 IF Z=1 THEN 10
840 END
```

Depreciation Rate

This program calculates the annual depreciation rate of an investment. You must provide the original price of the item, its resale price, and its age in years.

The depreciation rate is calculated by the following formula:

$$\text{depreciation rate} = 1 - \left(\frac{\text{resale price}}{\text{original price}}\right)^{1/\text{age}}$$

Example:

Joan bought her car for $4933.76 and sold it for $2400.00 three years later. What was its actual depreciation rate?

```
DEPRECIATION RATE

ORIGINAL PRICE? 4933.76
RESALE PRICE? 2400
YEARS? 3
DEPRECIATION RATE = 21.354 %

MORE DATA? (1=YES,0=NO)? 0
```

```
10 PRINT "□DEPRECIATION RATE"
20 PRINT
30 PRINT "ORIGINAL PRICE";
40 INPUT P
50 PRINT "RESALE PRICE";
60 INPUT T
70 PRINT "YEARS";
80 INPUT Y
89 REM - CALCULATE DEPRECIATION RATE BY FORMULA, CONVERT TO PERCENT
90 D=100*(1-(T/P)↑(1/Y))
100 PRINT "DEPRECIATION RATE =";
105 PRINT INT(1000*D+.5)/1000;"%"
110 PRINT
119 REM - RESTART OR END PROGRAM? USER INPUT REQUIRED
120 PRINT "MORE DATA? (1=YES,0=NO)";
130 INPUT X
140 IF X=1 THEN 20
150 END
```

Depreciation Amount

This program calculates the dollar amount depreciated within a given year for a depreciating investment. You must provide the original price of the investment, its depreciation rate, and the year of depreciation.

The depreciation amount is calculated by the following formula:

$$D = P \cdot i \cdot (1 - i)^{Y - 1}$$

where: D = depreciation amount
P = original price
i = depreciation rate
Y = year of depreciation

Examples:

Joan bought her car for $4933.76. Her model car depreciates at an average annual rate of 21%. What amount has the car depreciated in each of the first three years she has owned it?

Joan is also concerned about the depreciation of the tape deck in her car. It cost her $155.00 two years ago, and has a depreciation rate of 22%. How much will its value decline in year three?

```
DEPRECIATION AMOUNT

ORIGINAL PRICE? 4933.76
DEPRECIATION RATE? 21
--(ENTER YEAR=0 TO END)--
YEAR? 1
DEPRECIATION = $ 1036.09

YEAR? 2
DEPRECIATION = $ 818.51

YEAR? 3
DEPRECIATION = $ 646.62

YEAR? 0
MORE DATA? (1=YES,0=NO)? 1

ORIGINAL PRICE? 155
DEPRECIATION RATE? 22
--(ENTER YEAR=0 TO END)--
YEAR? 3
DEPRECIATION = $ 20.75

YEAR? 0
MORE DATA? (1=YES,0=NO)? 0
```

```
10 PRINT "□DEPRECIATION AMOUNT"
20 PRINT
30 PRINT "ORIGINAL PRICE";
40 INPUT P
50 PRINT "DEPRECIATION RATE";
60 INPUT I
69 REM - CONVERT FROM PERCENT TO DECIMAL
70 I=I/100
80 PRINT "--(ENTER YEAR=0 TO END)--"
90 PRINT "YEAR";
100 INPUT Y
109 REM - THROUGH CALCULATING FOR THIS ITEM?
110 IF Y=0 THEN 160
119 REM - CALCULATE DEPRECIATION AMOUNT BY FORMULA
120 D=P*I*(1-I)↑(Y-1)
129 REM - ROUND OFF TO NEAREST CENT, PRINT
130 PRINT "DEPRECIATION = $";
135 PRINT INT(D*100+.5)/100
140 PRINT
149 REM - RETURN FOR NEXT YEAR NUMBER
150 GOTO 90
159 REM - RESTART OR END PROGRAM? USER INPUT REQUIRED
160 PRINT "MORE DATA? (1=YES,0=NO)";
170 INPUT X
180 IF X=1 THEN 20
190 END
```

Salvage Value

This program calculates the salvage value of an item at the end of a given year. It is necessary for you to provide the age of the item, its original price, and its depreciation rate.

The salvage value is obtained by the following formula:

$$S = P(1 - i)^Y$$

where: S = salvage value
P = original price
i = depreciation rate
Y = age in years

Example:

What is the salvage value of Joan's car if it is three years old, she bought it for $4933.76, and it depreciates 21% annually? What would its salvage value be next year?

Joan's tape deck is 2 years old. What is its salvage value if it cost $155.00 originally and depreciates at a rate of 22%?

```
SALVAGE VALUE

ORIGINAL PRICE? 4933.76
DEPRECIATION RATE? 21
--(ENTER YEAR=0 TO END)--
YEARS? 3
VALUE = $ 2432.54

YEARS? 4
VALUE = $ 1921.7

YEARS? 0
MORE DATA? (1=YES,0=NO)? 1

ORIGINAL PRICE? 155
DEPRECIATION RATE? 22
--(ENTER YEAR=0 TO END)--
YEARS? 3
VALUE = $ 94.3

YEARS? 0
MORE DATA? (1=YES,0=NO)? 0
```

```
10 PRINT "SALVAGE VALUE"
20 PRINT
30 PRINT "ORIGINAL PRICE";
40 INPUT P
50 PRINT "DEPRECIATION RATE";
60 INPUT I
```

```
70 PRINT "--(ENTER YEAR=0 TO END)--"
80 PRINT "YEARS";
90 INPUT Y
99 REM - CALCULATE ANOTHER SALVAGE VALUE?
100 IF Y=0 THEN 140
108 REM - CALCULATE SALVAGE VALUE BY FORMULA, ROUND OFF, PRINT
109 REM - DEPRECIATION RATE CONVERTED TO DECIMAL FOR USE IN CALCULA
    TIONS
110 PRINT "VALUE = $";
115 PRINT INT(100*P*(1-I/100)↑Y+.5)/100
120 PRINT
129 REM - RETURN FOR NEXT YEAR NUMBER
130 GOTO 80
139 REM - RESTART OR END PROGRAM? USER INPUT REQUIRED
140 PRINT "MORE DATA? (1=YES,0=NO)";
150 INPUT X
160 IF X=1 THEN 20
170 END
```

Discount Commercial Paper

This program calculates the amount of discount and net cost of a discounted commercial paper. You must provide the future value of the paper, the discount rate and the number of days to maturity.

The formulas used to calculate the discount and cost are as follows:

$$\text{discount} = T \cdot \frac{D}{100} \cdot \frac{N}{360}$$

$$\text{cost} = T - \text{discount}$$

where: T = total future value
D = discount rate
N = number of days to maturity

Example:

Canning Corporation purchases a $625,000.00 commercial paper due in 60 days at 5.4%. What is the discount and cost?

```
DISCOUNT COMMERCIAL PAPER

FUTURE VALUE? 625000
DISCOUNT RATE? 5.4
DAYS TO MATURITY? 60
DISCOUNT = $ 5625
    COST = $ 619375

MORE DATA? (1=YES,0=NO)? 0
```

```
10 PRINT "DISCOUNT COMMERCIAL PAPER"
20 PRINT
29 REM - STATEMENTS 30 TO 90 REQUEST USER INPUT
30 PRINT "FUTURE VALUE";
40 INPUT T
50 PRINT "DISCOUNT RATE";
60 INPUT D
69 REM - CONVERT PERCENT TO DECIMAL
70 D=D/100
80 PRINT "DAYS TO MATURITY";
90 INPUT N
99 REM - CALACULATE DISCOUNT, PRINT
100 D1=T*D*N/360
110 PRINT "DISCOUNT = $";D1
119 REM - CALCULATE COST, PRINT
120 PRINT "    COST = $";T-D1
130 PRINT
139 REM - RESTART OR END PROGRAM? USER INPUT REQUIRED
```

```
140 PRINT "MORE DATA? (1=YES,0=NO)";
150 INPUT X
160 IF X=1 THEN 20
170 END
```

Principal on a Loan

This program calculates an initial amount borrowed. This amount is dependent upon the interest rate, the amount of regular payments, the number of payments per year and the term of the loan.

The calculation is based on the formula:

$$P = \frac{R \cdot N}{i} \cdot \left(1 - \frac{1}{(1 + i/N)^{N \cdot Y}}\right)$$

where: P = principal
R = regular payment
i = annual interest rate
N = number of payments per year
Y = number of years

Example:

Susan has agreed to pay $250.00 bimonthly for 3 years to repay a loan with 20% interest. What is the amount of the loan?

Tom can afford to make payments of $180.00 per month to repay a loan. If he is willing to make payments for four and a half years and the loan company charges 16% interest, what is the maximum amount Tom can borrow?

```
PRINCIPAL ON A LOAN

REGULAR PAYMENT? 250
TERM IN YEARS? 3
ANNUAL INTEREST RATE? 20
# OF PAYMENTS PER YEAR? 6
PRINCIPAL = $ 3343.45

MORE DATA? (1=YES,0=NO)? 1

REGULAR PAYMENT? 180
TERM IN YEARS? 4.5
ANNUAL INTEREST RATE? 16
# OF PAYMENTS PER YEAR? 12
PRINCIPAL = $ 6897.51

MORE DATA? (1=YES,0=NO)? 0
```

```
10 PRINT "PRINCIPAL ON A LOAN"
20 PRINT
29 REM - STATEMENTS 30 TO 100 REQUEST USER INPUT
30 PRINT "REGULAR PAYMENT";
40 INPUT R
```

```
50 PRINT "TERM IN YEARS";
60 INPUT Y
70 PRINT "ANNUAL INTEREST RATE";
80 INPUT I
90 PRINT "# OF PAYMENTS PER YEAR";
100 INPUT N
108 REM - CALCULATE AMOUNT OF PRINCIPAL BY FORMULA;
109 REM - INTEREST CONVERTED FROM PERCENT TO DECIMAL FOR CALCULATIONS
110 P=R*N*(1-1/((I/100)/N+1)↑(N*Y))/(I/100)
119 REM - ROUND OFF TO NEAREST CENT, PRINT
120 PRINT "PRINCIPAL = $";
125 PRINT INT(P*100+.5)/100
130 PRINT
139 REM - RESTART OR END PROGRAM? USER INPUT REQUIRED
140 PRINT "MORE DATA? (1=YES,0=NO)";
150 INPUT X
160 IF X=1 THEN 20
170 END
```

OPTION

In some cases it may be more convenient to enter the term of the loan in years and months rather than just years. The program changes necessary are listed following the example below.

Example:

What would be the amount of the mortgage if you were paying $75.00 a month for 11 months with 3% interest?

```
PRINCIPAL ON A LOAN

REGULAR PAYMENT? 75
TERM IN YEARS, MONTHS? 0,11
ANNUAL INTEREST RATE? 3
# OF PAYMENTS PER YEAR? 12
PRINCIPAL = $ 812.76

MORE DATA? (1=YES,0=NO)? 0
```

```
1 REM - OPTION 50-65
10 PRINT "□PRINCIPAL ON A LOAN"
 .
 .
 .
40 INPUT R
50 PRINT "TERM IN YEARS, MONTHS"
60 INPUT Y0,M
64 REM - CALCULATE YEARS FROM YEARS AND MONTHS
65 Y=(12*Y0+M)/12
70 PRINT "ANNUAL INTEREST RATE";
 .
 .
 .
170 END
```

Regular Payment on a Loan

This program calculates the amount required as regular payments in order to repay a loan over a specified time period. The specifications you must provide are the amount of the principal, the interest rate charged, the number of payments to be made per year and the number of years to pay. This program assumes all installment payments will be equal.

The calculation is based on the formula:

$$R = \frac{i \cdot P/N}{1 - \left(\frac{i}{N} + 1\right)^{-N \cdot Y}}$$

where: R = regular payment
i = annual interest rate
P = principal
N = number of payments per year
Y = number of years

Examples:

What must you pay on a loan of $4000.00 at 8% if payments are to be made quarterly for five years?

If Michael borrows $6500.00 at 12.5% from Best Rate Savings & Loan to be paid back over a period of 5.5 years, what would his monthly payments be?

```
REGULAR PAYMENT ON A LOAN

TERM IN YEARS? 5
PRINCIPAL? 4000
ANNUAL INTEREST RATE? 8
# OF PAYMENTS PER YEAR? 4
REGULAR PAYMENT = $ 244.63

MORE DATA? (1=YES,0=NO)? 1

TERM IN YEARS? 5.5
PRINCIPAL? 6500
ANNUAL INTEREST RATE? 12.5
# OF PAYMENTS PER YEAR? 12
REGULAR PAYMENT = $ 136.68

MORE DATA? (1=YES,0=NO)? 0
```

```
10 PRINT "REGULAR PAYMENT ON A LOAN"
20 PRINT
29 REM - STATEMENTS 30 TO 100 REQUEST USER INPUT
30 PRINT "TERM IN YEARS";
40 INPUT Y
50 PRINT "PRINCIPAL";
```

```
60 INPUT P
70 PRINT "ANNUAL INTEREST RATE";
80 INPUT I
90 PRINT "# OF PAYMENTS PER YEAR";
100 INPUT N
108 REM - CALCULATE AMOUNT OF REGULAR PAYMENT BY FORMULA;
109 REM - INTEREST CONVERTED FROM PERCENT TO DECIMAL FOR CALCULATIONS
110 R=((I/100)*P/N)/(1-1/((I/100)/N+1)↑(N*Y))
119 REM - ROUND OFF TO NEAREST CENT, PRINT
120 PRINT "REGULAR PAYMENT = $";
125 PRINT INT(R*100+.5)/100
130 PRINT
139 REM - RESTART OR END PROGRAM? USER INPUT REQUIRED
140 PRINT "MORE DATA? (1=YES,0=NO)";
150 INPUT X
160 IF X=1 THEN 20
170 END
```

OPTION

You may find it more convenient to enter the term of payment in years and months rather than years. The program changes necessary are listed following the example below.

Example:

Mr. Terry needs $10,000.00 to put down on a new home. Best Rates offers this amount at 14.0% interest to be repaid over a period of 11 years and 5 months. What would be the amount of regular monthly payments?

```
REGULAR PAYMENT ON A LOAN

TERM IN YEARS, MONTHS? 11,5
PRINCIPAL? 10000
ANNUAL INTEREST RATE? 14
# OF PAYMENTS PER YEAR? 12
REGULAR PAYMENT = $ 146.59

MORE DATA? (1=YES,0=NO)? 0
```

```
1 REM - OPTION 30-45
10 PRINT "REGULAR PAYMENT ON A LOAN"
 .
 .
 .
29 REM - STATEMENTS 30 TO 100 REQUEST USER INPUT
30 PRINT "TERM IN YEARS, MONTHS"
40 INPUT Y0,M
44 REM - CALCULATE YEARS FROM YEARS AND MONTHS
45 Y=(12*Y0+M)/12
50 PRINT "PRINCIPAL";
 .
 .
 .
170 END
```

Last Payment on a Loan

This program calculates the amount of the final payment on a loan. This final payment will complete amortization of a loan at the conclusion of its term. You must provide the amount of the loan, the amount of the regular payment, the interest rate charged, the number of payments per year and the term of payment.

The amount of the last payment is normally different from the amount of the regular payment. The final payment will be a "balloon" payment if the final payment is larger than the regular payment. A balloon payment is necessary if applying the amount of the regular payment as the last payment leaves a remaining balance due. In order to entirely pay off the loan at the end of its term, this remaining balance is added to the amount of the regular payment to determine the amount of the last payment.

On the other hand, the amount of the final payment is sometimes less than the regular payment. If the regular payment as the last payment would result in a negative loan balance, then the last payment should be smaller. In this case the regular payment is adjusted by the amount of this hypothetical negative balance to determine the amount of the last payment.

amount of last payment = regular payment + hypothetical balance due on a loan after $N \cdot Y$ regular payments

where: N = number of payments per year
Y = number of years

Examples:

Lynn borrowed $6000.00 at 5% from her father for college expenses. If she pays $1000.00 annually for seven years, what will her last payment be?

Lynn borrows $1150.00 at 8% interest to be repaid at a rate of $75.00 per month. A year and two months later Lynn decides to go to Europe. How much must she pay next month to completely pay off her loan?

```
LAST PAYMENT ON A LOAN

REGULAR PAYMENT? 1000
PRINCIPAL? 6000
TERM IN YEARS? 7
ANNUAL INTEREST RATE? 5
# OF PAYMENTS PER YEAR? 1
LAST PAYMENT = $ 1300.59

MORE DATA? (1=YES,0=NO)? 1

REGULAR PAYMENT? 75
PRINCIPAL? 1150
TERM IN YEARS? 1.17
ANNUAL INTEREST RATE? 8
# OF PAYMENTS PER YEAR? 12
LAST PAYMENT = $ 240.38

MORE DATA? (1=YES,0=NO)? 0
```

```
10 PRINT "LAST PAYMENT ON A LOAN"
20 PRINT
29 REM - STATEMENTS 30 TO 130 REQUEST USER INPUT
30 PRINT "REGULAR PAYMENT";
40 INPUT R
50 PRINT "PRINCIPAL";
60 INPUT P
70 PRINT "TERM IN YEARS";
80 INPUT Y
90 PRINT "ANNUAL INTEREST RATE";
100 INPUT I
109 REM - CONVERT INTEREST FROM PERCENT TO DECIMAL
110 I=I/100
120 PRINT "# OF PAYMENTS PER YEAR";
130 INPUT N
140 B0=P
149 REM - COMPUTE ALL PAYMENTS, BALANCES THROUGH LAST PAYMENT USING R
150 FOR J1=1 TO N*Y
159 REM - ROUND OFF INTEREST PAID TO NEAREST CENT
160 I1=INT((B0*I/N)*100+.5)/100
169 REM - CALCULATE AMOUNT AMORTIZED WITH EACH PAYMENT
170 A=R-I1
179 REM - BALANCE REMAINING DECREASES WITH EACH PAYMENT
180 B0=B0-A
190 NEXT J1
199 REM - CALCULATE LAST PAYMENT, ROUND OFF, PRINT
200 PRINT"LAST PAYMENT = $";
205 PRINT INT((R+B0)*100+.5)/100
210 PRINT
219 REM - RESTART OR END PROGRAM? USER INPUT REQUIRED
220 PRINT "MORE DATA? (1=YES,0=NO)";
230 INPUT X
240 IF X=1 THEN 20
250 END
```

OPTION

The program above allows the term of payment on the loan to be entered in years only. You may wish to enter the term in years and months instead. The program changes necessary are listed following the example.

Example:

If you pay $40.00 a month for 2 years and 3 months on a loan of $1200.00 at 7.5%, what amount will the last payment total?

```
LAST PAYMENT ON A LOAN

REGULAR PAYMENT? 40
PRINCIPAL? 1200
TERM IN YEARS, MONTHS? 2,3
ANNUAL INTEREST RATE? 7.5
# OF PAYMENTS PER YEAR? 12
LAST PAYMENT = $ 287.36

MORE DATA? (1=YES,0=NO)? 0
```

```
1 REM - OPTION 70-85
10 PRINT "LAST PAYMENT ON A LOAN"
 .
 .
 .
60 INPUT P
70 PRINT "TERM IN YEARS, MONTHS"
80 INPUT Y0,M
84 REM - CALCULATE YEARS FROM YEARS AND MONTHS
85 Y=(12*Y0+M)/12
90 PRINT "ANNUAL INTEREST RATE";
 .
 .
 .
250 END
```

Remaining Balance on a Loan

This program calculates the balance remaining on a loan after a specified number of payments. It is necessary for you to provide the amount of the regular payment, the number of payments per year, the amount of the principal, the annual interest rate, and the payment number after which to calculate the remaining balance.

The remaining balance is calculated by the following method:

$$\text{remaining balance} = \text{principal} - \text{amount amortized after } N \cdot (Y - 1) + N1 \text{ payments}$$

where: N = number of payments per year
Y = year to calculate remaining balance
$N1$ = payment number in year Y to calculate remaining balance

Example:

Kelly has taken out a loan of $8000.00 at 17.2% interest. His regular payments are $200.00 per month. If he has paid through the tenth payment in the fourth year, how much more does Kelly owe on his loan?

```
REMAINING BALANCE ON A LOAN

REGULAR PAYMENT? 200
PRINCIPAL? 8000
# OF PAYMENTS PER YEAR? 12
ANNUAL INTEREST RATE? 17.2
LAST PAYMENT MADE:
(PAYMENT NUMBER,YEAR)? 10,4
REMAINING BALANCE = $ 2496.17

MORE DATA? (1=YES,0=NO)? 0
```

```
10 PRINT "REMAINING BALANCE ON A LOAN"
20 PRINT
29 REM - STATEMENTS 30 TO 130 REQUEST USER INPUT
30 PRINT "REGULAR PAYMENT";
40 INPUT R
50 PRINT "PRINCIPAL";
60 INPUT P
70 PRINT "# OF PAYMENTS PER YEAR";
80 INPUT N
90 PRINT "ANNUAL INTEREST RATE";
100 INPUT I
109 REM - CONVERT FROM PERCENT TO DECIMAL
110 I=I/100
119 REM - ENTER THE PAYMENT NUMBER WITHIN THE YEAR, I.E. N1<=N
120 PRINT "LAST PAYMENT MADE:"
125 PRINT "(PAYMENT NUMBER,YEAR)";
130 INPUT N1,Y
139 REM - INITIALIZE REMAINING BALANCE
140 B0=P
```

```
149 REM - LOOP TO ACCUMULATE AMOUNT PAID SO FAR
150 FOR J1=1 TO N*(Y-1)+N1
159 REM - CALCULATE INTEREST PAID WITH EACH PAYMENT
160 I1=INT((B0*I/N)*100+.5)/100
169 REM - CALCULATE AMOUNT AMORTIZED WITH EACH PAYMENT
170 A=R-I1
179 REM - CALCULATE REMAINING BALANCE ON PRINCIPAL
180 B0=B0-A
190 NEXT J1
199 REM - ROUND OFF, PRINT
200 PRINT "REMAINING BALANCE = $";
205 PRINT INT(B0*100+.5)/100
210 PRINT
219 REM - RESTART OR END PROGRAM? USER INPUT REQUIRED
220 PRINT "MORE DATA? (1=YES,0=NO)";
230 INPUT X
240 IF X=1 THEN 20
250 END
```

OPTION

You may wish to specify the number of the last payment made as the total payment number rather than the payment number within a certain year. For instance, when 4 payments are made per year, payment 3 of year 3 would be entered as payment number 11. The program changes necessary are listed following the example below.

Example:

John made ten quarterly payments of $550.00 on a loan of $6000.00 with 16% interest. What is his remaining balance?

```
REMAINING BALANCE ON A LOAN

REGULAR PAYMENT? 550
PRINCIPAL? 6000
# OF PAYMENTS PER YEAR? 4
ANNUAL INTEREST RATE? 16
NUMBER OF PAYMENTS MADE? 10
REMAINING BALANCE = $ 24278.09

MORE DATA? (1=YES,0=NO)? 0
```

```
1 REM - OPTION 119-130, 150
10 PRINT "REMAINING BALANCE ON A LOAN"
 .
 .
 .
110 I=I/100
119 REM - ENTER THE PAYMENT NUMBER OF PAYMENTS MADE TO DATE
120 PRINT "NUMBER OF PAYMENTS MADE";
130 INPUT N1
139 REM - INITIALIZE REMAINING BALANCE
140 B0=P
149 REM - LOOP TO ACCUMULATE AMOUNT PAID SO FAR
150 FOR J1=1 TO N1
159 REM - CALCULATE INTEREST PAID WITH EACH PAYMENT
 .
 .
 .
250 END
```

Term of a Loan

This program calculates the period of time needed to repay a loan. You must specify the amount of the loan, the amount of the payments, the number of payments to be made per year and the annual interest rate on the loan. All payments are assumed to be equal.

The term of payment is derived from the following formula:

$$Y = -\frac{\log\left(1 - \frac{P \cdot i}{N \cdot R}\right)}{\log\left(1 + \frac{i}{N}\right)} \cdot \frac{1}{N}$$

where: Y = term of payment in years
P = principal
i = annual interest rate
N = number of payments per year
R = amount of payments

Examples:

What would be the duration of payment on a mortgage of $20,000.00 at 18% when payments of $1000.00 are to be made quarterly?

Sally takes out a loan for $12,669.00 at 16.8%. Her payments are $512.34 every two months. What is the term of her loan?

```
TERM OF A LOAN

REGULAR PAYMENT? 1000
PRINCIPAL? 20000
ANNUAL INTEREST RATE? 18
# OF PAYMENTS PER YEAR? 4
TERM = 13.1 YEARS

MORE DATA? (1=YES,0=NO)? 1

REGULAR PAYMENT? 512.34
PRINCIPAL? 12669
ANNUAL INTEREST RATE? 16.8
# OF PAYMENTS PER YEAR? 6
TERM = 7.1 YEARS

MORE DATA? (1=YES,0=NO)? 0
```

```
10 PRINT "□TERM OF A LOAN"
20 PRINT
29 REM - STATEMENTS 30 TO 100 REQUEST USER INPUT
30 PRINT "REGULAR PAYMENT";
40 INPUT R
50 PRINT "PRINCIPAL";
60 INPUT P
70 PRINT "ANNUAL INTEREST RATE";
80 INPUT I
90 PRINT "# OF PAYMENTS PER YEAR";
100 INPUT N
108 REM - CALCULATE TERM IN YEARS BY FORMULA;
109 REM - INTEREST CONVERTED FROM PERCENT TO DECIMAL FOR CALCULATION
110 Y=-(LOG(1-(P*(I/100))/(N*R))/(LOG(1+I/100/N)*N))
119 REM - ROUND OFF TO NEAREST TENTH, PRINT
120 PRINT "TERM =";INT(Y*10+.5)/10;
125 PRINT "YEARS"
130 PRINT
139 REM - RESTART OR END PROGRAM? USER INPUT REQUIRED
140 PRINT "MORE DATA? (1=YES,0=NO)";
150 INPUT X
160 IF X=1 THEN 20
170 END
```

OPTION

It is possible to calculate the term of payment in years and months rather than just years. To do this, make the program changes listed following the example below.

Example:

Dick took out a loan for $8000.00 at 7.5%. Regular payments of $150.00 are to be made monthly. How long will it take to pay off the loan?

```
TERM OF A LOAN

REGULAR PAYMENT? 150
PRINCIPAL? 8000
ANNUAL INTEREST RATE? 7.5
# OF PAYMENTS PER YEAR? 12
TERM = 5 YEARS, 5 MONTHS

MORE DATA? (1=YES,0=NO)? 0
```

```
1REM - OPTION 114-120
10 PRINT "□TERM OF A LOAN"
 .
 .
 .
110 Y=-(LOG(1-(P*(I/100))/(N*R))/(LOG(1+I/100/N)*N))
114 REM -CALCULATE YEARS AND MONTHS FROM YEARS
115 M=INT(Y*12+.5)
116 Y0=INT(M/12)
117 M=M-Y0*12
119 REM - PRINT RESULTS
120 PRINT "TERM =";Y0;"YEARS,";M;"MONTHS"
130 PRINT
 .
 .
 .
170 END
```

Annual Interest Rate on a Loan

This program calculates the rate at which interest is charged on a loan. To determine this rate you must enter the amount of the loan, the amount of the regular payment, the number of payments per year, and the term of the loan.

The annual interest rate is computed by the following method of approximation:

1) Guess an interest rate
   Initialize last guess to 0
2) Compute regular payment using guessed rate:

$$R_1 = \frac{i \cdot P/N}{1-((1+i)/N)^{-N \cdot Y}}$$

   Round off R_1
3) If computed payment = actual payment, then current guess = approximate interest rate
4) Otherwise, save current guess and calculate a new guess

$$i_2 = i$$

$$i = i \pm |(i-i_2)/2| \begin{cases} + \text{ if } R_1 < R \\ - \text{ if } R_1 > R \end{cases}$$

5) Go to 2

where:
- i = interest rate
- i_2 = previous interest rate
- R = input regular payment
- R_1 = computed regular payment
- P = principal
- N = number of payments per year
- Y = number of years

Examples:

Cindy borrowed $3000.00 from her friend George with an agreement to pay back $400.00 quarterly for 2 years. At what interest rate is she being charged?

To pay back a loan of $10,000.00 John contracted to make monthly payments of $120.00 for 9.5 years. At what rate is interest being charged?

```
ANNUAL INTEREST RATE ON A LOAN

REGULAR PAYMENT? 400
TERM IN YEARS? 2
PRINCIPAL? 3000
# OF PAYMENTS PER YEAR? 4
ANNUAL INTEREST RATE = 5.827 %

MORE DATA? (1=YES,0=NO)? 1
```

```
REGULAR PAYMENT? 120
TERM IN YEARS? 9.5
PRINCIPAL? 10000
# OF PAYMENTS PER YEAR? 12
ANNUAL INTEREST RATE = 6.933 %

MORE DATA? (1=YES,0=NO)? 0
```

```
10 PRINT "□ANNUAL INTEREST RATE ON A LOAN"
20 PRINT
29 REM - STATEMENTS 30 TO 100 REQUEST USER INPUT
30 PRINT "REGULAR PAYMENT";
40 INPUT R
50 PRINT "TERM IN YEARS";
60 INPUT Y
70 PRINT "PRINCIPAL";
80 INPUT P
90 PRINT "# OF PAYMENTS PER YEAR";
100 INPUT N
109 REM - GUESS AN INTEREST RATE (10%) TO INITIATE TESTING
110 I=10
119 REM - I2=LAST GUESS OR ESTIMATE (START WITH 0)
120 I2=0
129 REM - COMPUTE REGULAR PAYMENT USING GUESSED INTEREST RATE
130 R1=(I*P/N)/(1-1/((I/N+1)↑(N*Y)))
139 REM - ROUND OFF TO NEAREST CENT
140 R1=INT(R1*100+.5)/100
149 REM - I3=NUMBER USED TO CLOSE IN ON INTEREST RATE
150 I3=ABS(I-I2)/2
159 REM - SAVE THIS GUESS
160 I2=I
168 REM - COMPARE COMPUTED PAYMENT (R1) TO INPUT PAYMENT (R);
169 REM - IF THEY'RE EQUAL, LAST RATE GUESSED=APPROXIMATE INTEREST RATE
170 IF R1=R THEN 230
180 IF R1>R THEN 210
189 REM - R1<R, RATE MUST BE HIGHER THAN LAST GUESS
190 I=I+I3
199 REM - RETEST WITH NEW GUESS
200 GOTO 130
209 REM - R1>R, RATE MUST BE LOWER THAN LAST GUESS
210 REM
212 I=I-I3
219 REM - RETEST WITH NEW GUESS
220 GOTO 130
229 REM - COMPUTE INTEREST TO PROPER PROPORTIONS, ROUND OFF,PRINT
230 I=((INT((I*1000)*100+.5))/100)/1000
240 PRINT "ANNUAL INTEREST RATE =";
245 PRINT I*100;"%"
250 PRINT
259 REM - RESTART OR END PROGRAM? USER INPUT REQUIRED
260 PRINT "MORE DATA? (1=YES,0=NO)";
270 INPUT X
280 IF X=1 THEN 20
290 END
```

OPTION

The above listing allows the term of the loan to be entered in years only. You may wish to enter the term in years and months rather than years. The program changes necessary are listed following the example below.

Example:

If Connie pays $100.00 per month for 11 years and 7 months on a $10,000.00 loan, what is the annual interest rate on the loan?

```
ANNUAL INTEREST RATE ON A LOAN

REGULAR PAYMENT? 100
TERM IN YEARS, MONTHS? 11,7
PRINCIPAL? 10000
# OF PAYMENTS PER YEAR? 12
ANNUAL INTEREST RATE = 6.002 %

MORE DATA? (1=YES,0=NO)? 0
```

```
1REM - OPTION 50-65
10 PRINT "□ANNUAL INTEREST RATE ON A LOAN"
 .
 .
 .
40 INPUT R
50 PRINT "TERM IN YEARS, MONTHS";
60 INPUT Y0,M
64 REM - CALCULATE YEARS FROM YEARS AND MONTHS
65 Y=(12*Y0+M)/12
70 PRINT "PRINCIPAL";
 .
 .
 .
290 END
```

Mortgage Amortization Table

This program calculates and prints a loan repayment schedule. This schedule provides the following outputs:

1) Payment number
2) Amount of each payment paid as interest
3) Amount of the loan amortized with each payment
4) Balance remaining on the principal at the time of each payment
5) Accumulated interest paid at the time of each payment
6) Amount of the last payment

In addition, the yearly totals of interest paid and amount amortized are tabulated and printed.

To use this program you must supply the amount of the regular payment, the term of payment, the number of payments per year, the amount of the principal and the annual interest rate.

The schedule is calculated in the following manner:

1) Payment number = payment number within each year
2) Amount of each payment paid as interest = remaining balance $\cdot\, i/N$
   where: i = annual interest rate
   N = number of payments per year
3) Amount amortized with each payment = $R - I$
   where: R = amount of regular payment
   I = amount of each payment paid as interest
4) Balance remaining = $P - \Sigma A$
   where: P = principal
   ΣA = sum of amounts amortized with each payment to date
5) Accumulated interest = ΣI
   where: ΣI = sum of amounts of each payment paid as interest to date
6) Amount of last payment = $R + (P - R \cdot N \cdot Y)$
   where: R = regular payment
   P = principal
   N = number of payments per year
   Y = number of years

Example:

David needs $2100.00 to pay off some debts. His sister offers him the money at 6% interest. With payments of $75.00 monthly for 2½ years, what is David's repayment schedule?

```
MORTGAGE AMORTIZATION TABLE

REGULAR PAYMENT? 75
TERM IN YEARS? 2.5
PRINCIPAL? 2100
ANNUAL INTEREST RATE? 6
```

```
PAYMENTS PER YEAR? 12
START WITH WHAT YEAR? 1

ENTER 'C' TO CONTINUE? C


MORTGAGE AMORTIZATION TABLE

PRINCIPAL $ 2100  AT 6 %  FOR 2.5 YEARS
REGULAR PAYMENT = $ 75

 NO   INTR    AMORT    BALANCE    ACCUM INT
 1    10.5     64.5     2035.5     10.5
 2    10.18    64.82    1970.68    20.68
 3    9.85     65.15    1905.53    30.53
 4    9.53     65.47    1840.06    40.06
 5    9.2      65.8     1774.26    49.26
 6    8.87     66.13    1708.13    58.13
 7    8.54     66.46    1641.67    66.67
 8    8.21     66.79    1574.88    74.88
 9    7.87     67.13    1507.75    82.75
 10   7.54     67.46    1440.29    90.29
 11   7.2      67.8     1372.49    97.49
 12   6.86     68.14    1304.35    104.35

YR 1   104.35   795.65


ENTER 'C' TO CONTINUE? C


MORTGAGE AMORTIZATION TABLE

PRINCIPAL $ 2100  AT 6 %  FOR 2.5 YEARS
REGULAR PAYMENT = $ 75

 NO   INTR    AMORT    BALANCE    ACCUM INT
 1    6.52     68.48    1235.87    110.87
 2    6.18     68.82    1167.05    117.05
 3    5.84     69.16    1097.89    122.89
 4    5.49     69.51    1028.38    128.38
 5    5.14     69.86    958.52     133.52
 6    4.79     70.21    888.31     138.31
 7    4.44     70.56    817.75     142.75
 8    4.09     70.91    746.84     146.84
 9    3.73     71.27    675.57     150.57
 10   3.38     71.62    603.95     153.95
 11   3.02     71.98    531.97     156.97
 12   2.66     72.34    459.63     159.63

YR 2   55.28    844.72


ENTER 'C' TO CONTINUE? C
```

```
MORTGAGE AMORTIZATION TABLE

PRINCIPAL $ 2100  AT 6 %  FOR 2.5 YEARS
REGULAR PAYMENT = $ 75

 NO   INTR    AMORT    BALANCE    ACCUM INT
 1    2.3      72.7     386.93     161.93
 2    1.93     73.07    313.86     163.86
 3    1.57     73.43    240.43     165.43
 4    1.2      73.8     166.63     166.63
 5    .83      74.17    92.46      167.46
 6    .46      92.46    0          167.92
   LAST PAYMENT = $ 92.92

YR 3   8.29   459.63


CHANGE DATA AND RECOMPUTE?
(1=YES,0=NO)? 0
```

```
10 PRINT "□MORTGAGE AMORTIZATION TABLE"
20 PRINT
29 REM - STATEMENTS 30 TO 150 REQUEST USER INPUT
30 PRINT "REGULAR PAYMENT";
40 INPUT R
50 PRINT "TERM IN YEARS";
60 INPUT Y
70 PRINT "PRINCIPAL";
80 INPUT P
90 PRINT "ANNUAL INTEREST RATE";
100 INPUT I
109 REM - CONVERT FROM PERCENT TO DECIMAL
110 I=I/100
120 PRINT "PAYMENTS PER YEAR";
130 INPUT N
140 PRINT "START WITH WHAT YEAR";
150 INPUT X
159 REM - START PRINTING AT BEGINNING OF A YEAR
160 X=INT(X)
169 REM - INITIALIZE VARIABLES
170 C1=0
180 I2=0
190 I3=0
200 J0=0
210 N1=N
220 K=20
230 B0=P
240 A1=0
250 A2=0
259 REM - TERM LESS THAN ONE YEAR?
260 IF INT(Y)>=1 THEN 270
262 N1=((Y-INT(Y))*12)/12*N
264 J0=J0+1
265 GOTO 280
269 REM - LOOP FOR EACH YEAR
```

```
270 FOR J0=1 TO INT(Y)
279 REM - START PRINTING?
280 IF J0<X THEN 410
289 REM - CHECK FOR FULL SCREEN (20 LINES)
290 IF K+N+3<20 THEN 400
295 REM - WAIT FOR OPERATOR CUE TO GO TO NEXT SCREEN
297 PRINT
300 PRINT "ENTER 'C' TO CONTINUE";
310 INPUT W$
330 PRINT
339 REM - PRINT PAGE HEADINGS
340 PRINT "MORTGAGE AMORTIZATION TABLE":PRINT
350 PRINT "PRINCIPAL $";P;" AT";I*100;
355 PRINT "%  FOR";Y"YEARS"
360 PRINT "REGULAR PAYMENT = $";R
370 PRINT
380 PRINT " NO  INTR   AMORT",
385 PRINT "BALANCE","ACCUM INT"
389 REM - COUNT LINES PRINTED ON EACH PAGE IN K
390 K=7
400 K=K+N+3
410 FOR J1=1 TO N1
419 REM - CALCULATE INTEREST PAID THIS PAYMENT, ROUND OFF
420 I1=INT((B0*I/N)*100+.5)/100
429 REM - COUNT NUMBER OF PAYMENTS MADE SO FAR
430 C1=C1+1
439 REM - CALCULATE AMOUNT AMORTIZED THIS PAYMENT
440 A=R-I1
449 REM - SUM AMOUNT AMORTIZED TO DATE
450 A1=A1+A
459 REM - CALCULATE BALANCE DUE
460 B0=P-A1
468 REM - LAST PAYMENT? IF YES, CALCULATE AMOUNT SO THAT THE
469 REM - BALANCE DUE EQUALS $00.00 AFTER THIS PAYMENT
470 IF C1<>N*Y THEN 520
480 R=R+B0
490 A=A+B0
500 A1=A1+B0
510 B0=0
519 REM - SUM INTEREST PAID TO DATE
520 I2=I2+I1
529 REM - SUM INTEREST PAID THIS YEAR
530 I3=I3+I1
539 REM - SUM AMOUNT AMORTIZED THIS YEAR
540 A2=A2+A
541 A2=INT(A2*100+.5)/100
549 REM - STARTED PRINTING? IF YES, PRINT COMPUTED VALUES IN TABLE
550 IF J0<X THEN 570
551 A=INT(A*100+.5)/100
560 PRINT J1;TAB(4);I1;TAB(12);A;
561 B0=INT(B0*100+.5)/100
565 PRINT TAB(20);B0;TAB(28);I2
570 NEXT J1
579 REM - LAST PAYMENT? IF YES, ROUND OFF, PRINT
580 IF C1<>N*Y THEN 600
590 PRINT "   LAST PAYMENT = $";INT(R*     100+.5)/100
599 REM - STARTED PRINTING? IF YES,PRINT YEARLY TOTALS
```

```
600 IF J0<X THEN 640
610 PRINT
620 PRINT "YR";J0;I3;A2
630 PRINT
639 REM - COMPLETED TERM?
640 IF J0>Y THEN 720
649 REM - REINITIALIZE YEARLY VARIABLES
650 I3=0
660 A2=0
670 NEXT J0
671 J0=J0-1
679 REM - NEED TO PRINT A PARTIAL YEAR?
680 IF Y<>J0 THEN 262
720 PRINT
729 REM - RESTART OR END PROGRAM? USER INPUT REQUIRED
730 PRINT "CHANGE DATA AND RECOMPUTE?"
735 PRINT "(1=YES,0=NO)";
740 INPUT Z
750 IF Z=1 THEN 20
760 END
```

OPTION

You may wish to enter the term of payment in years and months rather than years. The program changes necessary are listed following the example below.

Example:

If you took out a loan for $700.00 from a friend at 9% interest and were to pay $100.00 per month for 8 months, what would your repayment schedule be?

```
MORTGAGE AMORTIZATION TABLE

REGULAR PAYMENT? 100
TERM IN YEARS, MONTHS? 0,8
PRINCIPAL? 700
ANNUAL INTEREST RATE? 9
PAYMENTS PER YEAR? 12
START WITH WHAT YEAR? 1


ENTER 'C' TO CONTINUE? C
```

```
MORTGAGE AMORTIZATION TABLE

PRINCIPAL $ 700  AT 9 %
      FOR 0 YEARS 8 MONTHS
REGULAR PAYMENT = $ 100

 NO   INTR    AMORT    BALANCE    ACCUM INT
 1    5.25     94.75    605.25     5.25
 2    4.54     95.46    509.79     9.79
 3    3.82     96.18    413.61     13.61
 4    3.1      96.9     316.71     16.71
 5    2.38     97.62    219.09     19.09
 6    1.64     98.36    120.73     20.73
 7    .91      99.09    21.64      21.64
 8    .16      21.64    0          21.8
   LAST PAYMENT = $ 21.8

YR 1  21.8    700


CHANGE DATA AND RECOMPUTE?
(1=YES,0=NO)?0
```

```
1 REM - OPTION 50-65,350
10 PRINT "□MORTGAGE AMORTIZATION TABLE"
 .
 .
 .
40 INPUT R
50 PRINT "TERM IN YEARS";
60 INPUT Y
64 REM - CONVERT YEARS AND MONTHS TO YEARS
65 Y=(12*Y0+M)/12
70 PRINT "PRINCIPAL";
 .
 .
 .
340 PRINT "MORTGAGE AMORTIZATION TABLE":PRINT
350 PRINT "PRINCIPAL $";P;" AT";I*100;
355 PRINT "%:"       FOR";Y0;"YEARS";M;"MONTHS"
360 PRINT "REGULAR PAYMENT = $";R
 .
 .
 .
760 END
```

Greatest Common Denominator

This program calculates the greatest common denominator of two integers. It is based on the Euclidean algorithm for finding the GCD:

1) Enter *A* , *B*
   A = absolute value of *A*
   B = absolute value of *B*
2) Calculate *R* = *A* - *B* • (integer of (*A*/*B*))
3) Is *R* = 0?. If yes, the GCD = *B*
   If no, go to step 4
4) *A* = *B*
   B = *R*
5) Go to step 2

Example:

Find the greatest common denominator of 50 and 18, 115 and 150.

```
GREATEST COMMON DENOMINATOR

(ENTER 0,0 TO END PROGRAM)
ENTER TWO NUMBERS? 50,18
G.C.D: 2

ENTER TWO NUMBERS? 115,150
G.C.D: 5

ENTER TWO NUMBERS? 0,0
```

```
10 PRINT "□GREATEST COMMON DENOMINATOR"
20 PRINT
30 PRINT "(ENTER 0,0 TO END PROGRAM)"
40 PRINT "ENTER TWO NUMBERS";
50 INPUT A,B
59 REM - END PROGRAM?
60 IF A<>0 THEN 90
70 IF B<>0 THEN 90
80 GOTO 190
89 REM - CALCULATE GCD ACCORDING TO EUCLIDEAN ALGORITHM, PRINT RESULT
90 A=ABS(A)
100 B=ABS(B)
110 R=A-B*INT(A/B)
120 IF R=0 THEN 160
130 A=B
140 B=R
150 GOTO 110
160 PRINT "G.C.D:";B
```

```
169 REM - PRINT BLANK LINE TO SEPARATE SETS OF DATA
170 PRINT
179 REM-RESTART PROGRAM
180 GOTO 40
190 END
```

Prime Factors of Integers

This program lists the prime factors of an integer. It will not test for the integer 0.

Examples:

What are the prime factors of -49?

Factor 92 into primes.

```
PRIME FACTORS OF INTEGERS

ENTER 0 TO END PROGRAM)
NUMBER? -49
-1
 7 ↑ 2

NUMBER? 92
 1
 2 ↑ 2
 23 ↑ 1

NUMBER? 0
```

```
10 PRINT "□PRIME FACTORS OF INTEGERS"
20 PRINT
30 PRINT "(ENTER 0 TO END PROGRAM)"
40 PRINT "NUMBER";
50 INPUT Z
59 REM - END PROGRAM?
60 IF Z=0 THEN 200
69 REM - THE SIGN OF THE NUMBER IS ALWAYS A FACTOR
70 PRINT SGN(Z)
79 REM - USE ABSOLUTE VALUE FOR CALCULATIONS
80 Z=ABS(Z)
88 REM - LOOP TO TEST ALL INTEGERS (2 THROUGH Z) AS PRIME FACTORS
89 REM - INTEGERS Z/2 THROUGH Z WILL HAVE NO NEW FACTORS
90 FOR I=2 TO Z/2
100 S=0
110 IF Z/I<>INT(Z/I) THEN 150
120 Z=Z/I
130 S=S+1
140 GOTO 110
149 REM - FIND A PRIME FACTOR? IF YES, PRINT
150 IF S=0 THEN 170
159 REM - PRINT FACTORS WITH EXPONENTS; I↑S=I TO THE S POWER
160 PRINT I;"↑";S
170 NEXT I
180 PRINT
190 GOTO 40
200 END
```

Area of a Polygon

This program calculates the area of a polygon. You must supply the x - and y -coordinates of all vertices. Coordinates must be entered in order of successive vertices.

The formula used to calculate the area is:

$$\text{Area} = (x_1 + x_2)\cdot(y_1 - y_2) + (x_2 + x_3)\cdot(y_2 - y_3) + \ldots (x_n + x_1)\cdot(y_n - y_1)\,12$$

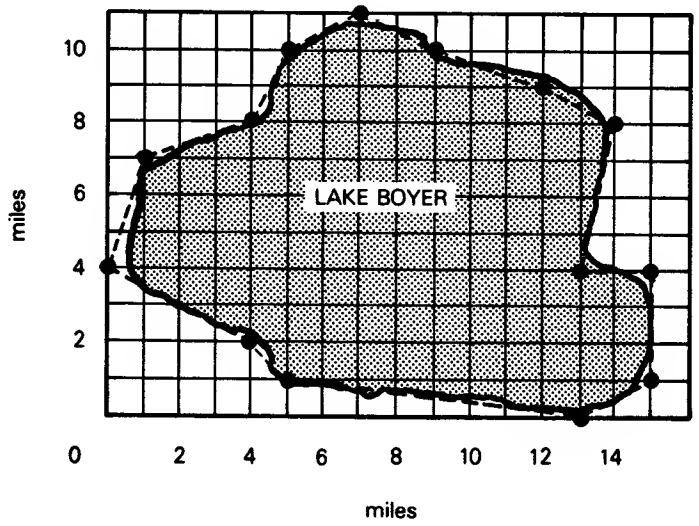
where n = the number of vertices.

The number of vertices you may enter is currently limited to 24. You may increase or decrease this limit by altering statement 30 according to the following scheme:

```
30 DIM X(n+1), Y(n+1)
```

Example:

Approximate the area of Lake Boyer.




```
AREA OF A POLYGON

(ENTER 0 TO END)
NUMBER OF VERTICES? 14
COORDINATES:
              VERTEX 1 ? 0,4
              VERTEX 2 ? 1,7
              VERTEX 3 ? 4,8
              VERTEX 4 ? 5,10
              VERTEX 5 ? 7,11
```

```
               VERTEX 6 ? 9,10
               VERTEX 7 ? 12,9
               VERTEX 8 ? 14,8
               VERTEX 9 ? 13,4
               VERTEX 10 ? 15,4
               VERTEX 11 ? 15,1
               VERTEX 12 ? 13,0
               VERTEX 13 ? 5,1
               VERTEX 14 ? 4,2
AREA = 108

(ENTER 0 TO END)
NUMBER OF VERTICES? 0
```

```
10 PRINT "□AREA OF A POLYGON"
20 PRINT
29 REM - COORDINATE ARRAYS SHOULD BE SET TO (NUMBER OF VERTICES +1)
30 DIM X(25),Y(25)
40 PRINT "(ENTER 0 TO END)"
45 PRINT "NUMBER OF VERTICES";
50 INPUT N
59 REM - END PROGRAM?
60 IF N=0 THEN 230
69 REM - LOOP TO ENTER COORDINATES IN ORDER OF SUCCESSIVE VERTICES
70 FOR I=1 TO N
80 IF I>1 THEN 110
90 PRINT "COORDINATES:"
95 PRINT "               VERTEX";I;
100 GOTO 120
110 PRINT "                VERTEX";I;
120 INPUT X(I),Y(I)
130 NEXT I
139 REM - FIRST VERTEX SERVES AS LAST VERTEX
140 X(N+1)=X(1)
150 Y(N+1)=Y(1)
160 A=0
169 REM - CALCULATE AREA, PRINT
170 FOR I=1 TO N
180 A=A+(X(I)+X(I+1))*(Y(I)-Y(I+1))
190 NEXT I
200 PRINT "AREA =";ABS(A)/2
210 PRINT
219 REM - RESTART PROGRAM
220 GOTO 40
230 END
```

Parts of a Triangle

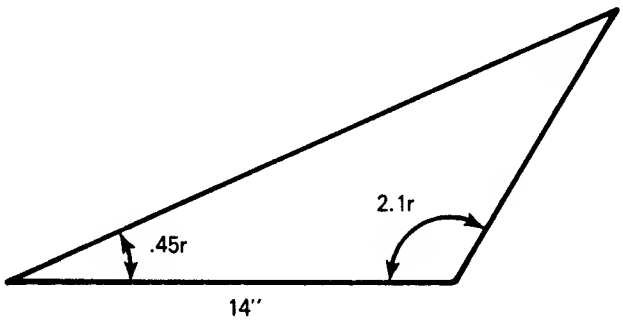
This program calculates three unknown parts of a triangle when three parts are given. At least one part given must be the length of a side. There are five possibilities for data entry:

1) Angle, side, angle
2) Side, angle, side
3) Angle, angle, side
4) Side, side, angle
5) Side, side, side

Data must be entered in the order it appears in a triangle, either clockwise or counterclockwise.

Example:

The base of a triangle measures 14 inches. The base angles measure .45 and 2.1 radians. What are the measurements of the triangle?




```
PARTS OF A TRIANGLE

PROBLEM TYPES:
                1=ASA, 2=SAS, 3=AAS
                4=SSA, 5=SSS, 6=END

ENTER PROBLEM TYPE? 1
ENTER ANGLE, SIDE, ANGLE? .45,14,2.1

SIDE 1 = 10.919
OPPOSITE ANGLE= .45 RADIANS
SIDE 2 = 21.67
OPPOSITE ANGLE= 2.1 RADIANS
SIDE 3 = 14
OPPOSITE ANGLE= .592 RADIANS

ENTER PROBLEM TYPE? 6
```

```
10 PRINT "■PARTS OF A TRIANGLE"
20 PRINT
30 DIM A(3),S(3)
38 REM - ENTER NUMBER OF PROBLEM TYPE ACCORDING TO KNOWN PARTS
39 REM - OF THE TRIANGLE WHERE A=ANGLE, S=LENGTH OF SIDE
40 PRINT "PROBLEM TYPES:"
50 PRINT SPC(15);"1=ASA, 2=SAS, 3=AAS"
55 PRINT SPC(15);"4=SSA, 5=SSS, 6=END"
57 PRINT
60 PRINT "ENTER PROBLEM TYPE";
70 INPUT X
79 REM - DIRECT PROGRAM TO PROPER CALCULATIONS
80 IF X=6 THEN 560
90 IF X=5 THEN 390
100 IF X=4 THEN 300
110 IF X=3 THEN 260
120 IF X=2 THEN 190
130 PRINT "ENTER ANGLE, SIDE, ANGLE";
140 INPUT A(1),S(3),A(2)
150 A(3)=π-A(1)-A(2)
160 S(1)=S(3)*SIN(A(1))/SIN(A(3))
170 S(2)=S(3)*SIN(A(2))/SIN(A(3))
180 GOTO 440
190 PRINT "ENTER SIDE, ANGLE, SIDE";
200 INPUT S(3),A(1),S(2)
210 S(1)=SQR(S(3)↑2+S(2)↑2-2*S(3)*S(2)*COS(A(1)))
220 A(2)=SIN(A(1))/S(1)*S(2)
230 A(2)=ATN(A(2)/SQR(1-(A(2))↑2))
240 A(3)=π-A(1)-A(2)
250 GOTO 440
260 PRINT "ENTER ANGLE, ANGLE, SIDE";
270 INPUT A(3),A(2),S(3)
280 A(1)=π-A(2)-A(3)
290 GOTO 160
300 PRINT "ENTER SIDE, SIDE, ANGLE";
310 INPUT S(1),S(2),A(1)
320 T=S(2)*SIN(A(1))
330 IF  S(1)<T THEN 520
340 S(3)=SQR(S(2)↑2-T↑2)
350 IF S(1)<=T THEN380
360 Y=SQR(S(1)↑2-T↑2)
370 S(3)=S(3)+Y
380 GOTO 220
390 PRINT "ENTER SIDE, SIDE, SIDE";
400 INPUT S(1),S(2),S(3)
410 A(1)=(S(2)↑2+S(3)↑2-S(1)↑2)/2/S(2)/S(3)
420 A(1)=ATN((SQR(1-(A(1))↑2))/A(1))
430 GOTO 220
440 PRINT
449 REM - RESTART PROGRAM
450 FOR I=1 TO 3
459 REM - THE ANGLE OF A TRIANGLE CANNOT BE LESS THAN ZERO
460 IF A(I)<0 THEN 520
470 PRINT "SIDE";I;"=";
475 PRINT INT(S(I)*1000+.5)/1000
480 PRINT "OPPOSITE ANGLE=";
```

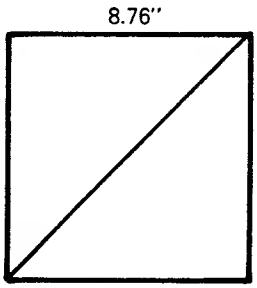
```
485 PRINT INT(A(I)*1000+.5)/1000;
486 PRINT "RADIANS"
490 NEXT I
500 PRINT
510 GOTO 60
520 PRINT
530 PRINT "NO SOLUTION"
540 PRINT
549 REM - RESTART PROGRAM
550 GOTO 60
560 END
```

OPTION

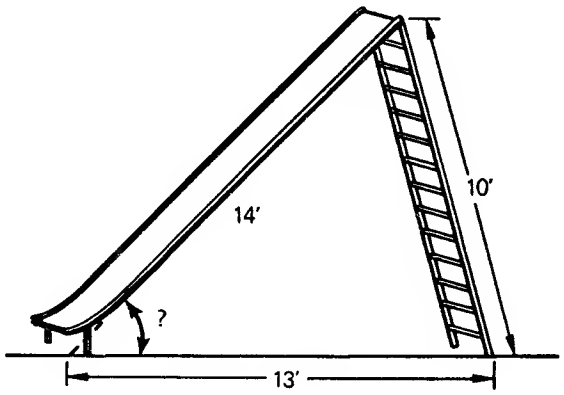
It may be more convenient for you to work with angles in degrees rather than radians. The program changes necessary are listed following the examples below.

Examples:

A square measures 8.76" x 8.76". What is the length of its diagonal?



The ladder of a slide measures 10', the slide 14', and it covers 13' of ground from base of ladder to tip of slide. How steep is the slide?




```
PARTS OF A TRIANGLE

PROBLEM TYPES:
                1=ASA, 2=SAS, 3=AAS
                4=SSA, 5=SSS, 6=END

ENTER PROBLEM TYPE? 2
ENTER ANGLE, SIDE, ANGLE? 8.76,90,8.76
```

```
SIDE 1 = 12.389
OPPOSITE ANGLE= 90 DEGREES
SIDE 2 = 8.76
OPPOSITE ANGLE= 45 DEGREES
SIDE 3 = 8.76
OPPOSITE ANGLE= 45 DEGREES

ENTER PROBLEM TYPE? 5
ENTER ANGLE, SIDE, ANGLE? 10,13,14

SIDE 1 = 10
OPPOSITE ANGLE= 43.279 DEGREES
SIDE 2 = 13
OPPOSITE ANGLE= 63.027 DEGREES
SIDE 3 = 14
OPPOSITE ANGLE= 73.694 DEGREES

ENTER PROBLEM TYPE? 6
```

```
 1 REM - OPTION 34-35,145-456,205,275-276,315,485
 10 PRINT "□PARTS OF A TRIANGLE"


 34 REM - SET CONVERSION FACTOR FOR CONVERTING DEGREES TO RADIANS
 35 C=.0174532927
 40 PRINT "PROBLEM TYPES"
  :
  :
140 INPUT A(1),S(3),A(2)
145 A(1)=A(1)*C
146 A(2)=A(2)*C
150 A(3)=π-A(1)-A(2)
  :
  :
200 INPUT S(3),A(1),S(2)
205 A(1)=A(1)*C
210 S(1)=SQR(S(3)↑2+S(2)↑2-2*S(3)*S(2)*COS(A(1)))
  :
  :
270 INPUT A(3),A(2),S(3)
275 A(3)=A(3)*C
276 A(2)=A(2)*C
280 A(1)=π-A(2)-A(3)
  :
  :
310 INPUT S(1),S(2),A(1)
315 A(1)=A(1)*C
320 T=S(2)*SIN(A(1))
  :
  :
470 PRINT "SIDE";I;"=";
475 PRINT INT(S(I)*1000+.5)/1000
480 PRINT "OPPOSITE ANGLE=";
485 PRINT "DEGREES"
490 NEXT I
  :
  :
560 END
```

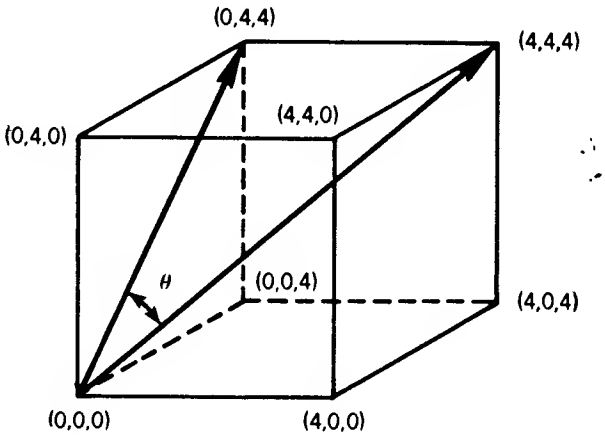
Analysis of Two Vectors

vektor berekening

This program calculates the angle between two given vectors, the angle between each vector and axis, and the magnitude of each vector. The vectors are given in three dimensional space.

Example:

Find the angle (θ) between a diagonal of a cube and a diagonal of one of its faces. The cube measures 4 x 4 x 4.




```
ANALYSIS OF TWO VECTORS

VECTOR 1: X,Y,Z? 0,4,4
VECTOR 2: X,Y,Z? 4,4,4

VECTOR 1
MAGNITUDE: 5.65685425
ANGLE WITH X-AXIS: 90.0000008
ANGLE WITH Y-AXIS: 45.0000004
ANGLE WITH Z-AXIS: 45.0000004

VECTOR 2
MAGNITUDE: 6.92820323
ANGLE WITH X-AXIS: 54.7356108
ANGLE WITH Y-AXIS: 54.7356108
ANGLE WITH Z-AXIS: 54.7356108

ANGLE BETWEEN VECTORS: 35.26439

MORE DATA (1=YES,0=NO)? 0
```

```
10 PRINT "□ANALYSIS OF TWO VECTORS"
20 PRINT
30 DIM X(2),Y(2),Z(2),M(2)
39 REM - STATMENTS 40 TO 70 REQUEST VECTOR COORDINATES
40 PRINT "VECTOR 1: X,Y,Z";
```

```
50 INPUT X(1),Y(1),Z(1)
60 PRINT "VECTOR 2: X,Y,Z";
70 INPUT X(2),Y(2),Z(2)
80 PRINT
89 REM - LOOP TO ANALYZE VECTORS
90 FOR I=1 TO 2
99 REM - CALCULATE MAGNITUDE, PRINT
100 M(I)=SQR(X(I)↑2+Y(I)↑2+Z(I)↑2)
109 REM - IS VECTOR A POINT?  IF YES, CANNOT COMPUTE AN ANGLE
110 IF M(I)=0 THEN 220
120 PRINT "VECTOR";I
130 PRINT "MAGNITUDE:";M(I)
139 REM - CONVERSION FACTOR FOR RADIANS TO DEGREES
140 S=57.29578
149 REM - CALCULATE ANGLE BETWEEN VECTOR AND X-AXIS, PRINT
150 J=X(I)/M(I)
160 PRINT "ANGLE WITH X-AXIS:";
162 IF J=0 THEN 167
165 PRINT ATN((SQR(1-J↑2))/J)*S
166 GOTO 170
167 PRINT S*π/2
169 REM - CALCULATE ANGLE BETWEEN VECTOR AND Y-AXIS, PRINT
170 J=Y(I)/M(I)
180 PRINT "ANGLE WITH Y-AXIS:";
182 IF J=0 THEN 187
185 PRINT ATN((SQR(1-J↑2))/J)*S
186 GOTO 190
187 PRINT S*π/2
189 REM - CALCULATE ANGLE BETWEEN VECTOR AND Z-AXIS, PRINT
190 J=Z(I)/M(I)
200 PRINT "ANGLE WITH Z-AXIS:";
202 IF J=0 THEN 207
205 PRINT ATN((SQR(1-J↑2))/J)*S
206 GOTO 210
207 PRINT S*π/2
210 PRINT
220 NEXT I
230 J=0
239 REM - IF EITHER VECTOR A POINT, CANNOT COMPUTE ANGLE
240 IF M(1)=0 THEN 310
250 IF M(2)=0 THEN 310
259 REM - CALCULATE ANGLE BETWEEN VECTORS
260 J=(X(1)*X(2)+Y(1)*Y(2)+Z(1)*Z(2))/M(1)/M(2)
269 REM - ARE VECTORS PERPENDICULAR?
270 IF J<>0 THEN 300
280 J=90
290 GOTO 310
299 REM - CALCULATE ANGLE IN DEGREES, PRINT
300 J=ATN(SQR(1-J↑2)/J)*S
310 PRINT "ANGLE BETWEEN VECTORS:";J
320 PRINT
329 REM - RESTART OR END PROGRAM? USER INPUT REQUIRED
330 PRINT "MORE DATA (1=YES,0=NO)";
340 INPUT Z
350 IF Z=1 THEN 40
360 END
```

Operations on Two Vectors

This program performs four operations on two vectors given in three space. The operations performed are:

1) Addition
2) Subtraction
3) Scalar (dot) product
4) Cross product

Example:

Vectors are drawn from the origin to two points A(5,-1,2) and B(1,4,9). Add, subtract, and find the dot and cross product of these vectors.

```
OPERATIONS ON TWO VECTORS

VECTOR A: X,Y,Z? 5,-1,2
VECTOR B: X,Y,Z? 1,4,9

A+B= 6 , 3 , 11
A-B= 4 ,-5 ,-7
A.B= 19
A*B=-17 ;-43 , 21

MORE DATA? (1=YES,0=NO)? 0
```

```
10 PRINT "□OPERATIONS ON TWO VECTORS"
20 PRINT
30 PRINT "VECTOR A: X,Y,Z";
40 INPUT X1,Y1,Z1
50 PRINT "VECTOR B: X,Y,Z";
60 INPUT X2,Y2,Z2
70 PRINT
79 REM - PERFORM VECTOR ADDITION, PRINT RESULTING VECTOR COORDINATES
80 PRINT "A+B=";X1+X2;",";Y1+Y2;
85 PRINT ",";Z1+Z2
89 REM - PERFORM VECTOR SUBTRACTION, PRINT RESULTING VECTOR COORDINATES
90 PRINT "A-B=";X1-X2;",";Y1-Y2;
95 PRINT ",";Z1-Z2
99 REM - CALCULATE DOT PRODUCT, PRINT
100 PRINT "A.B=";X1*X2+Y1*Y2+Z1*Z2
109 REM - CALCULATE CROSS PRODUCT, PRINT RESULTING VECTOR COORDINATES
110 PRINT "A*B=";Y1*Z2-Z1*Y2;";";
115 PRINT Z1*X2-X1*Z2;",";X1*Y2-Y1*X2
120 PRINT
129 REM - RESTART OR END PROGRAM? USER INPUT REQUIRED
130 PRINT "MORE DATA? (1=YES,0=NO)";
140 INPUT Z
150 IF Z=1 THEN 20
160 END
```

Angle Conversion: Radians to Degrees

This program converts an angle given in radians to degrees, minutes and seconds.

Example:

How many degrees, minutes and seconds are there in an angle of 2.5 radians? In 118 radians?

```
ANGLE CONVERSION: RADIANS TO DEGREES

(ENTER 0 TO END THIS PROGRAM)
ANGLE IN RADIANS? 2.5

  DEGREES = 143
  MINUTES = 14
  SECONDS = 22.02

ANGLE IN RADIANS? 118

  DEGREES = 280
  MINUTES = 54
  SECONDS = 7.15

  ANGLE IN RADIANS? 0
```

```
10 PRINT"ANGLE CONVERSION: RADIANS TO DEGREES
20 PRINT
30 PRINT "(ENTER 0 TO END THIS PROGRAM)"
50 PRINT "ANGLE IN RADIANS";
60 INPUT R
69 REM - TEST FOR END OF PROGRAM
70 IF R=0 THEN 170
79 REM - CONVERT RADIANS TO SECONDS
80 A=3600*180*R/π
89 REM - CALCULATE NUMBER OF WHOLE DEGREES
90 D=INT(A/3600)
99 REM - CALCULATE NUMBER OF FULL CIRCLES
100 D1=INT(D/360)
105 PRINT
109 REM - CALCULATE DEGREES OF ANGLE WITHIN 360 DEGREES, PRINT
110 PRINT "  DEGREES =";D-360*D1
119 REM - CALCULATE MINUTES, PRINT
120 PRINT "  MINUTES =";
125 PRINT INT((A-D*3600)/60)
129 REM - CALCULATE SECONDS, ROUND OFF, PRINT
130 S=A-D*3600-(INT((A-D*3600)/60))*60
140 PRINT "  SECONDS =";
145 PRINT INT(100*S+.5)/100
```

```
150 PRINT
159 REM - RESTART PROGRAM
160 GOTO 50
170 END
```

OPTION

You may prefer your answer in degrees and decimals of degrees rather than degrees, minutes and seconds. The program changes necessary are listed following the example below.

Example:

How many degrees are there in an angle of 2.5 radians?

```
ANGLE CONVERSION: RADIANS TO DEGREES

(ENTER 0 TO END THIS PROGRAM)
ANGLE IN RADIANS? 2.5

  DEGREES = 143

  ANGLE IN RADIANS? 0
```

```
1 REM - OPTION 110
10 PRINT"□ANGLE CONVERSION: RADIANS TO DEGREES
 .
 .
 .
109 REM - CALCULATE DEGREES OF ANGLE WITHIN 360 DEGREES, PRINT
110 PRINT "  DEGREES =";
111 PRINT INT((D-360*D1)*100+.5)/100
150 PRINT
 .
 .
 .
170 END
```

Angle Conversion: Degrees to Radians

This program converts an angle given in degrees, minutes and seconds to radians.

Examples:

An angle measures 30 degrees, 5 minutes and 3 seconds. What would be the measure of this angle in radians?

What would be the radian measurement of two angles measuring 278°, 19', 54'' and 721°, 0', 0''?

```
ANGLE CONVERSION: DEGREES TO RADIANS

(TO END, ENTER 0,0,0)
ANGLE IN DEGREES, MINUTES, SECONDS:
? 30,5,3
RADIANS = .525067685

ANGLE IN DEGREES, MINUTES, SECONDS:
? 278,19,54
RADIANS = 4.8578033

ANGLE IN DEGREES, MINUTES, SECONDS:
? 721,0,0
RADIANS = .0174514931

ANGLE IN DEGREES, MINUTES, SECONDS:
? 0,0,0
```

```
10 PRINT "□ANGLE CONVERSION: DEGREES TO RADIANS
20 PRINT
30 PRINT "(TO END, ENTER 0,0,0)"
40 PRINT "ANGLE IN DEGREES, ";
45 PRINT "MINUTES, SECONDS:"
50 INPUT D,M,S
59 REM - TEST FOR END OF PROGRAM
60 IF D<>0 THEN 100
70 IF M<>0 THEN 100
80 IF S<>0 THEN 100
90 GOTO 150
99 REM - CONVERT DEGREES, MINUTES, SECONDS TO DEGREES
100 A=D+M/60+S/3600
109 REM - CALCULATE NUMBER OF COMPLETE CIRCLES
110 R=INT(A/360)
119 REM - CALCULATE ANGLE WITHIN 360 DEGEES, PRINT
120 PRINT "RADIANS =";
125 PRINT A*.01745329-R*6.2831853
130 PRINT
139 REM - RESTART PROGRAM
140 GOTO 40
150 END
```

OPTION

It may be more convenient for you to enter the angle in degrees and fractions of degrees rather than degrees, minutes and seconds. The program changes necessary are listed following the example below.

Example:

How many radians are in an angle measuring 33.08°? 90°?

```
ANGLE CONVERSION: DEGREES TO RADIANS

(TO END, ENTER 0,0,0)
ANGLE IN DEGREES:? 33.08
RADIANS = .577354833

ANGLE IN DEGREES:? 90
RADIANS = 1.5707961

ANGLE IN DEGREES:? 0
```

```
1 REM - OPTION 30-60
10 PRINT "□ANGLE CONVERSION: DEGREES TO RADIANS
20 PRINT
30 PRINT "(TO END, ENTER 0,0,0)"
40 PRINT "ANGLE IN DEGREES, ";
50 INPUT A
59 REM - TEST FOR END OF PROGRAM
60 IF A=0 THEN 150
 .
 .
 .
109 REM - CALCULATE NUMBER OF COMPLETE CIRCLES
 .
 .
 .
150 END
```

Coordinate Conversion

This program converts the coordinates of a point given in Cartesian coordinates to polar coordinates, and vice versa.

The formulas for the conversions are:

$$r = \sqrt{x^2 + y^2}$$

$$A = \text{arctangent}\ (y/x)$$

$$x = r \cdot \text{cosine}\ (A)$$

$$y = r \cdot \text{sine}\ (A)$$

where: x = abscissa, y = ordinate } Cartesian coordinates

r = magnitude of ray, A = angle (in degrees) } polar coordinates

Examples:

Find Cartesian coordinates of the point (2,30.5°) given in polar coordinates.

If a point is at (7,18) in the Cartesian system, what are its coordinates in the polar system?

A point is located at (0,-46.8). What is its location in polar coordinates?

```
COORDINATE CONVERSION

    ( 1=CARTESIAN TO POLAR)
    (-1=POLAR TO CARTESIAN)
    ( 0=END PROGRAM)

WHICH DIRECTION? -1
R,A? 2,30.5
X = 1.72 , Y = 1.02

WHICH DIRECTION? 1
X,Y? 7,18
R = 19.31 , A = 68.75

WHICH DIRECTION? 1
X,Y? 0,-46.8
R = 46.8 , A = 270

WHICH DIRECTION? 0
```

```
10 PRINT "□COORDINATE CONVERSION"
20 PRINT
30 PRINT "     ( 1=CARTESIAN TO POLAR)"
40 PRINT "     (-1=POLAR TO CARTESIAN)"
50 PRINT "     ( 0=END PROGRAM)"
55 PRINT
60 PRINT "WHICH DIRECTION";
70 INPUT D
79 REM - END PROGRAM?
80 IF D=0 THEN 380
89 REM - DIRECT PROGRAM TO PERFORM PROPER CONVERSION
90 IF D=-1 THEN 320
98 REM - CONVERT FROM CARTESIAN COORDINATES TO POLAR COORDINATES
99 REM - ENTER CARTESIAN COORDINATES (ABSCISSA, ORDINATE)
100 PRINT "X,Y";
110 INPUT X,Y
119 REM - POINT ON Y-AXIS?
120 IF X=0 THEN 170
129 REM - POINT ON X-AXIS?
130 IF Y=0 THEN 260
139 REM - COMPUTE POLAR COORDINATES, ROUND OFF, PRINT
140 PRINT "R =";INT(SGN(X)*SQR(X↑2+Y↑2)*100+.5)/100;",";
150 PRINT " A =";INT(ATN(Y/X)*180/π*100+.5)/100
160 GOTO 55
169 REM - POINT IS ON Y-AXIS; AT ORIGIN?
170 IF Y=0 THEN 240
180 PRINT "R =";ABS(Y);",";
189 REM - IS POINT ABOVE OR BELOW ORIGIN?
190 IF Y<0 THEN 220
200 PRINT " A = 90"
210 GOTO 55
220 PRINT " A = 270"
230 GOTO 55
239 REM - POINT IS AT ORIGIN
240 PRINT "R = 0,  A = 0"
250 GOTO 55
259 REM - POINT IS ON X-AXIS
260 PRINT "R =";ABS(X);",";
269 REM - IS POINT TO LEFT OR RIGHT OF ORIGIN?
270 IF X<0 THEN 300
280 PRINT " A = 0"
290 GOTO 55
300 PRINT " A = 180"
310 GOTO 55
318 REM - CONVERT FROM POLAR COORDINATES TO CARTESIAN COORDINATES
319 REM - ENTER POLAR COORDINATES (MAGNITUDE OF RAY, ANGLE)
320 PRINT "R,A";
330 INPUT R,A
339 REM - CONVERT FROM DEGREES TO RADIANS
340 M=(A-INT(A/360)*360)*π/180
349 REM - CALCULATE CARTESIAN COORDINATES, ROUND OFF, PRINT
350 PRINT "X =";
355 PRINT INT(R*COS(M)*100+.5)/100;
360 PRINT ", Y =";
365 PRINT INT(R*SIN(M)*100+.5)/100
369 REM - RESTART PROGRAM
370 GOTO 55
380 END
```

Coordinate Plot

This program plots points on a set of coordinate axes. You must provide the *x* - and *y* -coordinates of all points to be plotted, the endpoints of the *x* - and *y* -axes, and the increment between points on each axis.

The graph is unconventional in that its *x* -axis runs vertically while its *y* -axis runs horizontally. In addition, the axes do not necessarily intersect at zero. A reminder as to where the axes intersect is printed at the top of each graph.

The limit on the number of points plotted may be increased or decreased by altering statement 30 in the following manner:

```
30 DIM X(N+1),Y(N+1)
```

where *N* = the maximum number of points you wish to plot.

The length of the y-axis is limited by the width of the output device. This program tests for a length not to exceed the width of the PET screen, 40 spaces. The test at statement 90 may be altered to accommodate the 80-column CBM screen. If using a CBM with an 80-column screen you might enter:

```
90 IF B2<=78 THEN 120
```

Example:

The heights of twelve men and their sons are recorded in the table below. Plot the data points.

father	65	63	67	64	68	62	70	66	68	67	69	71
son	68	66	68	65	69	66	68	65	71	67	68	70

height in inches

```
COORDINATE PLOT

X-AXIS: LOWER ENDPOINT,
UPPER ENDPOINT, INCREMENT? 22,33,.5
Y-AXIS: LEFT ENDPOINT,
RIGHT ENDPOINT, INCREMENT? 22,33,.5
NUMBER OF POINTS? 12
COORDINATES OF POINT  1 ? 25,28
                POINT  2 ? 23,26
                POINT  3 ? 27,28
                POINT  4 ? 24,25
                POINT  5 ? 28,29
                POINT  6 ? 22,26
                POINT  7 ? 30,28
                POINT  8 ? 26,25
                POINT  9 ? 28,31
                POINT  10 ? 27,27
                POINT  11 ? 29,28
                POINT  12 ? 31,30
```

```
INTERSECTION OF AXES AT ( 22 , 22 )

* * * * * * * * + * * * * * * * * * * * * * * * * Y
*
*                 +
*
*             +
*
*                         +
*
*           +
*
*                     +   +
*
*                             +         +
*
*                         +
*
*                         +
*
*                               +
*
*
*
*
*
X
```

```
10 PRINT "□COORDINATE PLOT"
20 PRINT
28 REM - DIMENSION OF X() AND Y() SHOULD BE LIMITED TO (N+1);
29 REM - WHERE N=THE NUMBER OF POINTS BEING PLOTTED, MAXIMUM LIMIT 99
30 DIM X(100),Y(100)
39 REM - INPUT INFORMATION TO SET UP AXES
40 PRINT "X-AXIS: LOWER ENDPOINT,"
45 PRINT "UPPER ENDPOINT, INCREMENT";
50 INPUT A1,A2,A3
60 PRINT "Y-AXIS: LEFT ENDPOINT,"
65 PRINT "RIGHT ENDPOINT, INCREMENT";
70 INPUT B1,B2,B3
80 B2=(B2-B1)/B3
88 REM - Y-AXIS TOO LONG FOR OUTPUT DEVICE? IF YES, CHANGE ENDPOINTS
89 REM - OR INCREASE INCREMENT
90 IF B2<=40 THEN 120
100 PRINT "Y-RANGE TOO LARGE"
110 GOTO 60
120 PRINT "NUMBER OF POINTS";
130 INPUT N
139 REM - NO POINTS TO PLOT? END PROGRAM
140 IF N=0 THEN 1070
149 REM - TOO MANY POINTS? IF YES, REENTER NUMBER OF POINTS
150 IF N<=99 THEN 180
160 PRINT "TOO MANY POINTS"
170 GOTO 120
179 REM - LOOP TO INPUT X,Y COORDINATES FOR EACH POINT
180 FOR I=1 TO N
190 IF I>1 THEN 220
```

```
200 PRINT "COORDINATES OF POINT ";I;
210 GOTO 230
220 PRINT "                    POINT ";I;
230 INPUT X(I), Y(I)
239 REM - ROUND OFF EACH X,Y TO NEAREST INCREMENT ON AXIS
240 X(I)= INT((X(I)-A1)/A3+.5)
250 Y(I)= INT((Y(I)-B1)/B3+.5)
260 NEXT I
269 REM - CALCULATE ADDITIONAL X AND Y COORDINATE
270 Y(N+1)=INT(B2+.5)+1
280 X(N+1)=INT((A2-A1)/A3+.5)+1
290 PRINT
299 REM - NOTE WHERE AXES CROSS
300 PRINT "INTERSECTION OF AXES AT (";
301 PRINT A1;",";B1;")"
310 PRINT
319 REM - SORT COORDINATES; REORDER X(1) TO X(N) SMALLEST TO LARGEST
320 FOR J=1 TO N
330 FOR I=1 TO N-J
340 A=X(I)
350 B=Y(I)
360 C=X(I+1)
370 D=Y(I+1)
380 IF A<C THEN 430
390 X(I)=C
400 Y(I)=D
410 X(I+1)=A
420 Y(I+1)=B
430 NEXT I
440 NEXT J
449 REM - NEXT POINT TO BE PLOTTED STORED IN T
450 T=1
459 REM - SKIP POINTS OUT OF X-POSITIVE RANGE
460 FOR P=0 TO N-1
470 IF X(P+1)>=0 THEN Q=P:P=N
480 NEXT P
485 P=Q
489 REM - LOOP TO CALL UP EACH X-INCREMENT FOR LINES OF PRINT
490 FOR I=0 TO INT((A2-A1)/A3+.5)
500 T=T+P
509 REM - COUNT NUMBER OF POINTS TO BE PLOTTED ON EACH LINE IN P
510 P=0
519 REM - ALL POINTS PLOTTED?
520 IF T>N THEN 540
529 REM - X-VALUE ON X-LINE? IF YES, TEST FOR Y
530 IF X(T)=I THEN 590
539 REM - FIRST LINE? IF YES, Y-AXIS MUST BE PLOTTED
540 IF I=0 THEN 570
549 REM - PLOT X-AXIS
550 PRINT "*";
560 GOTO 1040
570 S=N+1
580 GOTO 920
590 FOR L=T TO N
599 REM - NEXT POINT PLOTTED ON SAME LINE
600 IF X(L)<=X(T) THEN P=P+1
620 NEXT L
```

```
629 REM - PLOT ONE POINT
630 IF P=1 THEN 730
638 REM - LOOP TO SORT Y-COORDINATES WITH EQUAL X-COORDINATES;
639 REM - REORDER SMALLEST TO LARGEST
640 FOR J=1 TO P
650 FOR L=1 TO P-J
660 D=Y(T+L-1)
670 B=Y(T+L)
680 IFD<=B THEN 710
690 Y(T+L-1)=B
700 Y(T+L)=D
710 NEXT L
720 NEXT J
730 FOR L=0 TO P-1
740 Z=Y(T+L)
749 REM - TEST FOR OUT-OF RANGE Y-COORDINATE
750 IF Z>=0 THEN 770
760 NEXT L
769 REM - POINT TO BE PLOTTED ON X-AXIS?
770 IF I=0 THEN 910
779 REM - POINT TO BE PLOTTED ON Y-AXIS?
780 IF Z=0 THEN 800
789 REM - PLOT X-AXIS
790 PRINT"*";
800 IF L=P-1 THEN 870
810 FOR J=L TO P-1
819 REM - TEST FOR OUT-OF RANGE Y-COORDINATE
820 IF Z>B2 THEN 1040
829 REM - BYPASS DUPLICATE COORDINATES
830 IF Y(T+J)=Z THEN 860
839 REM - PLOT POINT
840 PRINT TAB(Z);"+";
850 Z=Y(T+J)
860 NEXT J
869 REM - TEST FOR OUT-OF RANGE Y-COORDINATE
870 IF Z<0 THEN 1040
880 IF Z>B2 THEN 1040
889 REM - PLOT POINT
890 PRINT TAB(Z);"+";
900 GOTO 1040
910 S=T+L
919 REM - LOOP TO ESTABLISH PRINT FOR FIRST LINE
920 FOR J=0 TO B2
929 REM - POINT TO BE PLOTTED?
930 IF Y(S)<>J THEN 1010
939 REM - PLOT POINT
940 PRINT"+";
949 REM - BYPASS DUPLICATE COORDINATES
950 FOR K=S TO T+P-1
960 IF Y(K)=Y(S) THEN 990
970 S=K
980 GOTO 1020
990 NEXT K
1000 GOTO 1020
1009 REM - PLOT Y-AXIS
1010 PRINT"*";
1020 NEXT J
```

```
1029 REM - LABEL Y-AXIS
1030 PRINT"Y";
1039 REM - ADVANCE OUTPUT DEVICE TO NEXT LINE
1040 PRINT
1050 NEXT I
1059 REM - LABEL X-AXIS
1060 PRINT"X"
1070 END
```

Plot of Polar Equation

This program plots a given function in polar coordinates. There are up to 90 points plotted, one every four degrees. (Some points may overlap.)

The graph is conventional in that the x-axis runs horizontally, the y-axis runs vertically, and they intersect at zero. You need only specify the absolute value of the endpoints.

The increment between each point on the x- and y-axes is adjusted so that a value of one on either axis is equidistant from zero. This allows the function to be plotted with minimal distortion. An adjustment of each increment is necessary because of different spacing horizontally and vertically on the PET screen.

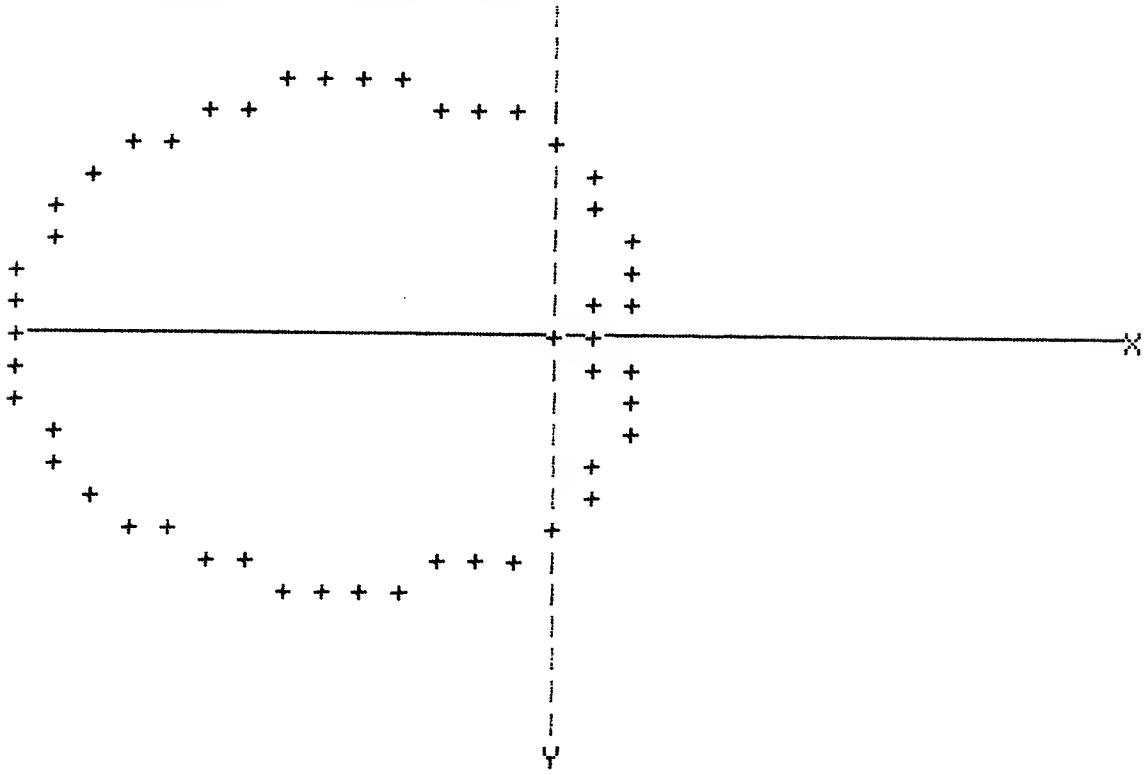
It is necessary for you to enter the function to be plotted before you run the program. The function must be entered as a function of d. $f(d)$ will be entered and set equal to F at line 130. For example, the function $f(d) = 2 \cdot (1-\cos(d))$ will be entered as follows:

```
130 F=2*(1-COS(D))
```

Example:

Plot the equation $f(d) = 2 \cdot (1-\cos(d))$.

```
130 F=2*(1-COS(D))
RUN


PLOT OF POLAR EQUATION

ABSOLUTE VALUE OF ENDPOINTS? 4

INCREMENT OF X-AXIS = .285714286
INCREMENT OF Y-AXIS = .333333333

*** PLEASE WAIT A FEW MINUTES
FOR THE GRAPH TO APPEAR ***
```



```
10 PRINT "PLOT OF POLAR EQUATION"
20 PRINT
28 REM - COORDINATE ARRAYS SET FOR 90 POINTS;
29 REM - ONE EXTRA X-COORDINATE IS CALCULATED IN PROGRAM
30 DIM X(91),Y(90)
39 REM - NUMBER OF POINTS TO BE CALCULATED
40 N=90
49 REM - ABSOLUTE VALYUE OF ALL ENDPOINTS ARE EQUAL
50 PRINT "ABSOLUTE VALUE OF ENDPOINTS";
60 INPUT Z
70 PRINT
75 X1=14
76 Y1=12
79 REM - CALCULATE INCREMENTS OF AXES ACCORDING TO CHARACTERS PER AXIS
80 PRINT "INCREMENT OF X-AXIS =";Z/X1
90 PRINT "INCREMENT OF Y-AXIS =";Z/Y1
100 PRINT
104 PRINT "*** PLEASE WAIT A ";
105 PRINT "FEW MINUTES"
106 PRINT "FOR THE GRAPH TO APPEAR***"
110 FOR I=1 TO N
119 REM - CONVERT DEGREES TO RADIANS
120 D=.06981317*I
129 REM - ENTER FUCTION HERE (F-"FUNCTION")
130 PRINT "ENTER FUCTION AT LINE 130!"
138 REM - CALCULATE EACH CARTESIAN COORDINATE,
139 REM - ROUND OFF TO NEAREST INCREMENT ON AXIS
140 X(I)=INT(((F*COS(D)/Z+1)*X1)+.5)
150 Y(I)=INT(((-F*SIN(D)/Z+1)*Y1)+.5)
160 NEXT I
169 REM - SORT COORDINATES; REORDER Y(1) TO Y(N) SMALLEST TO LARGEST
170 FOR J=1 TO N
180 FOR I=1 TO N-J
190 A=X(I)
200 B=Y(I)
210 IF B<=Y(I+1) THEN 260
220 X(I)=X(I+1)
230 Y(I)=Y(I+1)
240 X(I+1)=A
250 Y(I+1)=B
260 NEXT I
270 NEXT J
279 REM - NEXT POINT TO BE PLOTTED STORED INT
280 T=1
289 REM - SKIP POINTS OUT OF Y-POSITIVE RANGE
290 FOR P=0 TO N-1
300 IF Y(P+1)>=0 THEN 320
310 NEXT P
319 REM - LOOP TO CALL UP EACH Y-INCREMENT FOR LINES OF PRINT
320 FOR I=0 TO Y1*2
330 T=T+P
339 REM - NUMBER OF POINTS TO BE PLOTTED ON EACH LINE STORED IN P
340 P=0
349 REM - ALL POINTS PLOTTED?
350 IF T>N THEN 370
359 REM - Y-VALUE ON Y-LINE?
360 IF Y(T)=I THEN 420
```

```
369 REM - PRINT X-AXIS
370 IF I=Y1 THEN 400
379 REM - PRINT Y-AXIS
380 PRINT TAB(X1);"|";
390 GOTO 860
400 S=N+1
410 GOTO 740
420 FOR L=T TO N
429 REM - NEXT POINT TO BE PLOTTED ON SAME LINE?
430 IF Y(L)>Y(T) THEN 450
440 P=P+1
450 NEXT L
460 IF P=1 THEN 560
468 REM - LOOP TO SORT X-COORDINATES WITH EQUAL Y-COORDINATES;
469 REM - REORDER SMALLEST TO LARGEST
470 FOR J=1 TO P
480 FOR L=1 TO P-J
490 C=X(T+L-1)
500 A=X(T+L)
510 IF C<=A THEN 540
520 X(T+L-1)=A
530 X(T+L)=C
540 NEXT L
550 NEXT J
559 REM - PRINT X-AXIS?
560 IF I=Y1 THEN 730
570 L=-1
580 S=0
590 FOR K=0 TO P-1
599 REM - MORE THAN ONE POINT TO BE PLOTTED AT SAME POINT ON GRAPH?
600 IF X(T+K)=L THEN 690
610 L=X(T+K)
619 REM - PLOT POINT TO THE LEFT OF Y-AXIS?
620 IF L=X1 THEN 660
630 IF L<X1 THEN 670
640 IF S=1 THEN 670
649 REM - PRINT Y-AXIS
650 PRINT TAB(X1);"|";
660 S=1
669 REM - POINT OUTSIDE OF POSITIVE X-RANGE?
670 IF L>X1*2 THEN 860
679 REM - PLOT POINT
680 PRINT TAB(L);"+";
690 NEXT K
700 IF S=1 THEN 860
709 REM - PRINT Y-AXIS
710 PRINT TAB(X1);"|";
720 GOTO 860
730 S=T
739 REM - LOOP TO PRINT LINE OF X-AXIS
740 FOR J=0 TO X1*2
750 IF X(S)<>J THEN 830
759 REM - PLOT POINT ON X-AXIS
760 PRINT "+";
770 FOR K=S TO T+P-1
780 IF X(K)=X(S) THEN 810
790 S=K
```

```
800 GOTO 840
810 NEXT K
820 GOTO 840
829 REM - PRINT X-AXIS
830 PRINT "-";
840 NEXT J
849 REM - LABEL X-AXIS
850 PRINT "X";
860 PRINT
870 NEXT I
879 REM - LABEL Y-AXIS
880 PRINT TAB(X1);"Y"
890 END
```

Plot of Functions

This program calculates and plots up to nine functions. All functions must be functions of x, and all will be plotted on the same set of axes.

To set up the axes you must input the endpoints of the x - and y -axes. You must also state the increment by which the points on each axis are to be increased.

The graph is unconventional in that its x -axis runs vertically while its y -axis runs horizontally. To read the graph you must either turn your output 90° counterclockwise or mentally adjust to the change in convention.

The graph is also unconventional in that its axes do not necessarily cross at zero. A reminder as to where the axes cross is printed at the top of each graph.

You must enter the functions to be plotted as program statements prior to running the program. Statement numbers 221 to 229 are reserved for this purpose. Functions must be entered in the number sequence Y(1), Y(2),. . . Y(9). For example, if you wish to plot the functions $f(x) = 2x + 1$ and $f(x) = \sqrt{x}$, you must type:

```
221 Y(1)=2*X+1
222 Y(2)=SQR(X)
```

The length of the y-axis is limited by the width of your output device. This program tests for a length not to exceed the width of the PET screen, 40 spaces. The test at statement 140 may be altered to accommodate the 80-column CBM screen. For example, an 80-column CBM screen will accommodate a graph 78 spaces wide by changing statement 140 to:

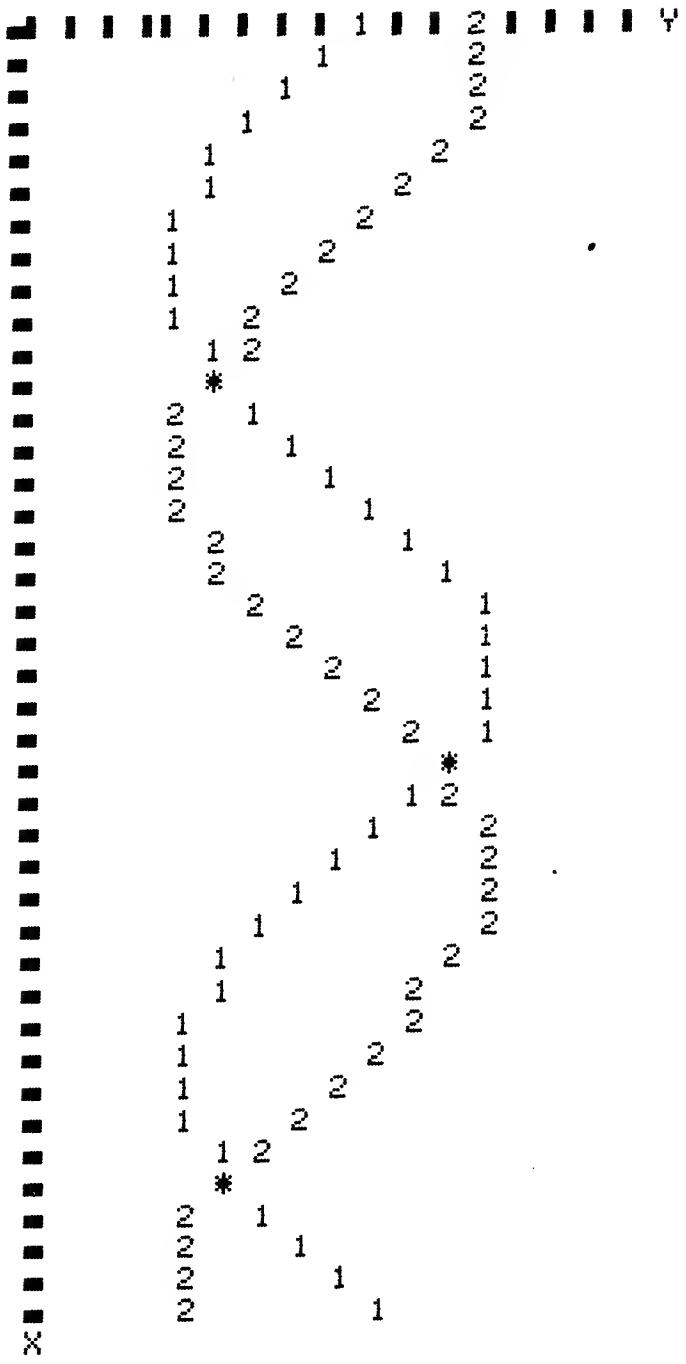
```
140 IF Y2<=78 THEN 170
```

Example:

Plot the equations $f(x) = \cos(x)$ and $f(x) = \sin(x)$.

```
221 Y(1)=COS(X)
222 Y(2)=SIN(X)
RUN

PLOT OF FUNCTIONS

NUMBER OF FUNCTIONS? 2
X-AXIS: LOWER ENDPOINT,
UPPER ENDPOINT,INCREMENT? -5,5,.25
Y-AXIS: LEFT ENDPOINT,
RIGHT ENDPOINT, INCREMENT? -2,2,.25
```

X AXIS CROSSES Y-AXIS AT Y=-2
Y AXIS CROSSES X-AXIS AT X=-5



```
10 PRINT "PLOT OF FUNCTIONS"
20 PRINT
29 REM - NUMBER OF FUNCTIONS WHICH CAN BE PLOTTED IS LIMITED TO 9
30 DIM Y(9),A$(11)
40 FOR I=1 TO 11
49 REM - GET VALUES FOR A$-ARRAY FROM DATA TABLE AT STATEMENTS 470-471
50 READ A$(I)
60 NEXT I
69 REM - STATEMENTS 79 TO 120 REQUEST USER INPUT
70 PRINT "NUMBER OF FUNCTIONS";
80 INPUT N
90 PRINT "X-AXIS: LOWER ENDPOINT,"
95 PRINT "UPPER ENDPOINT,INCREMENT";
100 INPUT X1,X2,X3
110 PRINT "Y-AXIS: LEFT ENDPOINT,"
115 PRINT "RIGHT ENDPOINT, INCREMENT";
120 INPUT Y1,Y2,Y3
129 REM - CALCULATE NUMBER OF SPACES ON Y-AXIS
130 Y2=(Y2-Y1)/Y3
138 REM - TEST FOR A Y-AXIS TOO LONG FOR OUTPUT DEVICE.
139 REM - IF YES; THEN LESSEN RANGE OR INCREASE INCREMENT
140 IF Y2<=36 THEN 170
150 PRINT "Y-RANGE TOO LARGE"
160 GOTO 110
170 PRINT
180 PRINT
189 REM - MAKE NOTE OF WHERE AXES CROSS
190 PRINT "X-AXIS CROSSES Y-AXIS ";
195 PRINT "AT Y=";Y1
200 PRINT "Y AXIS CROSSES X-AXIS ";
210 PRINT "AT X=";X1:PRINT
219 REM - SET UP LOOP TO READ VALUE AT EACH X-INCREMENT
220 FOR X=X1 TO X2 STEP X3
221 REM - FUNCTIONS Y(1) TO Y(9) SHOULD BE ENTERED AT LINES 221 TO 229
230 FOR I=1 TO N
239 REM - ESTABLISH THE ROUNDED VALUE OF Y FOR EACH X-INCREMENT VALUE
240 Y(I)= INT((Y(I)-Y1)/Y3+.5)
250 NEXT I
259 REM - LOOP TO READ VALUE OF EACH Y-INCREMENT
260 FOR I=0 TO Y2
269 REM - S COUNTS THE NUMBER OF VALUES AT EACH Y-INCREMENT FOR EACH X
270 S=0
280 FOR J=1 TO N
289 REM - PLOT A POINT ON THIS SPOT? IF YES, STORE FUNCTION NUMBER IN T
290 IF Y(J)<>I THEN 320
300 S=S+1
310 T=J
320 NEXT J
327 REM - TEST FOR NUMBER OF POINTS TO PLOT ON EACH SPOT;
328 REM - IF 0 PRINT "+" (FIRST LINE ONLY), IF 1 PRINT FUNCTION NUMBER,
329 REM - IF 2 OR MORE PRINT "*"
330 IF S>0 THEN 360
340 PRINT A$(SGN(I)+10);
350 GOTO 400
360 IF S>1 THEN 390
370 PRINT A$ (T);
380 GOTO 400
```

```
390 PRINT "*";
400 NEXT I
409 REM - LABEL AXES AT THE LAST SPACE ON EACH AXES
410 IF X>X1 THEN 430
420 PRINT "Y";
429 REM - ADVANCE PRINTER TO NEXT LINE
430 PRINT
439 REM - PRINT SPACE INSTEAD OF "+" AFTER FIRST LINE OF PRINT (Y-AXIS)
440 A$(11)=" "
450 NEXT X
460 PRINT "X"
470 DATA "1","2","3","4","5","6","7"
471 DATA "8","9","■","▮"
480 END
```

Linear Interpolation

This program calculates the y-coordinates of points on a line given their x-coordinates. It is necessary to know coordinates of two points on the same line.

The point is interpolated using the following formula:

$$y = y_1 + \frac{(y_2 - y_1) \cdot (x - x_1)}{(x_2 - x_1)}$$

where: x_1, y_1 = coordinates of first point on the line
x_2, y_2 = coordinates of second point on the line
x = abscissa of point to be interpolated
y = ordinate of the point on the line with x

. Examples:

A conversion table lists 60°F as 15.56°C and 90°F as 32.22°C. Calculate degrees Celsius of 73°F and 85.6°F.

A new sales tax of 17.5% has been imposed on us. What will be the tax on a sofa which sells for $455.68?

```
LINEAR INTERPOLATION

X,Y OF FIRST POINT? 60,15.56
X,Y OF SECOND POINT? 90,32.22
INTERPOLATE:X = ? 73
            Y =   22.779

MORE POINTS (1=YES,0=NO)? 1

INTERPOLATE:X = ? 85.6
            Y =   29.777

MORE POINTS (1=YES,0=NO)? 0

NEW LINE (1=YES,0=NO)? 1

X,Y OF FIRST POINT? 0,0
X,Y OF SECOND POINT? 100,17.5
INTERPOLATE:X = ? 455.68
            Y =   79.744

MORE POINTS (1=YES,0=NO)? 0

NEW LINE (1=YES,0=NO)? 0
```

```
10 PRINT "LINEAR INTERPOLATION"
20 PRINT
29 REM - ENTER X- AND Y-COORDINATES OF TWO POINTS ON THE LINE
30 PRINT "X,Y OF FIRST POINT";
40 INPUT X1,Y1
50 PRINT "X,Y OF SECOND POINT";
60 INPUT X2,Y2
69 REM - ENTER X-COORDINATE OF POINT TO BE INTERPOLATED
70 PRINT "INTERPOLATE:X = ";
80 INPUT X
89 REM - COMPUTE CORRESPONDING Y-COORDINATE
90 Y=Y1+(Y2-Y1)/(X2-X1)*(X-X1)
99 REM - ROUND OFF, PRINT
100 PRINT TAB(12);"Y =  ";
101 PRINT INT(Y*1000+.5)/1000
110 PRINT
120 PRINT "MORE POINTS (1=YES,0=NO)";
130 INPUT Z
140 PRINT
150 IF Z=1 THEN 70
159 REM - INTERPOLATE ON ANOTHER LINE?
160 PRINT "NEW LINE (1=YES,0=NO)";
170 INPUT Z
180 IF Z=1 THEN 20
190 END
```

Curvilinear Interpolation

This program computes y-coordinates of points on a curve given their x-coordinates. You must input coordinates of known points on the curve, no two having the same abscissa.

The computations are performed using the Lagrange method of interpolation.

The number of known points on the curve which may be entered in the program is limited to 50. You may increase or decrease this limit by altering statement 30 according to the following scheme:

```
30 DIM X(P), Y(P)
```

where P = the number of known points on a curve.

Examples:

Consider the curve $y = x^3 - 3x + 3$. You know that the points (-3,-15), (-2,1), (-1,5), (0,3), (1,1), (2,5), and (3,21) are on the curve. What is the value of y when $x = -1.65$ and 0.2?

Given the following points from a sine curve, what is the sine of -2.47 and the sine of 1.5?

(-5,.958)	(0,0)
(-4,.757)	(1,.841)
(-3,-.141)	(2,.909)
(-2,-.909)	(3,.141)
(-1,-.841)	(4,-.757)
	(5,-.959)

```
CURVILINEAR INTERPOLATION

NUMBER OF KNOWN POINTS? 7
X,Y OF POINT 1 ? -3,-15
X,Y OF POINT 2 ? -2,1
X,Y OF POINT 3 ? -1,5
X,Y OF POINT 4 ? 0,3
X,Y OF POINT 5 ? 1,1
X,Y OF POINT 6 ? 2,5
X,Y OF POINT 7 ? 3,21

INTERPOLATE: X= ? -1.65
             Y=   3.457875

MORE POINTS HERE (1=YES,0=NO)? 1

INTERPOLATE: X= ? .2
             Y=   2.408

MORE POINTS HERE (1=YES,0=NO)? 1
ANOTHER CURVE (1=YES,0=NO)? 1
NUMBER OF KNOWN POINTS? 11
X,Y OF POINT 1 ? -5,.958
X,Y OF POINT 2 ? -4,.757
X,Y OF POINT 3 ? -3,-.141
```

```
X,Y OF POINT 4 ? -2,-.909
X,Y OF POINT 5 ? -1,-.841
X,Y OF POINT 6 ? 0,0
X,Y OF POINT 7 ? 1,.841
X,Y OF POINT 8 ? 2,.909
X,Y OF POINT 9 ? 3,.141
X,Y OF POINT 10 ? 4,-.757
X,Y OF POINT 11 ? 5,-.959

INTERPOLATE: X= ? -2.47
             Y=   -.621839596

MORE POINTS HERE (1=YES,0=NO)? 1

INTERPOLATE: X= ? 1.5
             Y=   .9971638

MORE POINTS HERE (1=YES,0=NO)? 0
ANOTHER CURVE (1=YES,0=NO)? 0
```

```
10 PRINT "□CURVILINEAR INTERPOLATION"
20 PRINT
30 DIM X(50),Y(50)
40 PRINT "NUMBER OF KNOWN POINTS";
50 INPUT P
60 FOR I=1 TO P
70 PRINT "X,Y OF POINT";I;
80 INPUT X(I),Y(I)
90 NEXT I
100 PRINT
110 PRINT "INTERPOLATE: X= ";
120 INPUT A
130 B=0
140 FOR J=1 TO P
150 T=1
160 FOR I=1 TO P
170 IF I=J THEN 190
180 T=T*(A-X(I))/(X(J)-X(I))
190 NEXT I
200 B=B+T*Y(J)
210 NEXT J
220 PRINT "              Y=  ";B
230 PRINT
240 PRINT "MORE POINTS HERE?";
245 PRINT "(1=YES,0=NO)";
250 INPUT C
260 IF C=1 THEN 100
270 PRINT "ANOTHER CURVE? (1=YES,0=NO)";
280 INPUT C
290 IF C=1 THEN 40
300 END
```

Integration: Simpson's Rule

This program approximates the definite integral of a function. The integral is computed using Simpson's rule.

The method the program takes is optional: you must supply either the function of the curve or values of the function at specified intervals. For both methods you must enter the limits of integration and the increment between points within the limits.

If the function to be integrated is known, it must be entered before running the program. The function will be defined at line 50. For example, the function $f(x) = x^3$ will be entered as follows:

```
50 DEFFNC(X)=X↑3
```

Examples:

Find the definite integral of the function $f(x) = x^3$ between 0 and 2 with increments of .2 and .1.

What is the integral of a curve between -1 and 1 if the points known are as follows:

(-1,.54)	(.25,.969)
(-.75,.73)	(.5,.878)
(-.5,.878)	(.75,.73)
(-.25,.969)	(1,.54)
(0,1)	

```
50 DEFFNC(X)=X↑3
RUN

INTEGRATION: SIMPSON'S RULE

FORMULA:(1=KNOWN,0=UNKNOWN)? 1
THE LOWER, UPPER LIMITS? 0,2
INCREMENT OF X? .2
INTEGRAL IS 4

RUN


INTEGRATION: SIMPSON'S RULE

FORMULA:(1=KNOWN,0=UNKNOWN)? 1
THE LOWER, UPPER LIMITS? 0,2
INCREMENT OF X? .1
INTEGRAL IS 4

RUN


INTEGRATION: SIMPSON'S RULE

FORMULA:(1=KNOWN,0=UNKNOWN)? 0
THE LOWER, UPPER LIMITS? -1,1
INCREMENT OF X? .25
```

```
FIRST, LAST VALUE OF F(X)? .54,.54
VALUE OF F(X) AT:
          INTERVAL 1 (X=-.75 )? .73
VALUE OF F(X) AT:
          INTERVAL 2 (X=-.5 )? .878
VALUE OF F(X) AT:
          INTERVAL 3 (X=-.25 )? .969
VALUE OF F(X) AT:
          INTERVAL 4 (X= 0 )? 1
VALUE OF F(X) AT:
          INTERVAL 5 (X= 25 )? .969
VALUE OF F(X) AT:
          INTERVAL 6 (X= .5 )? .878
VALUE OF F(X) AT:
          INTERVAL 7 (X= .75 )? .73
 INTEGRAL IS 1.682
```

```
10 PRINT "□INTEGRATION: SIMPSON'S RULE"
20 PRINT
30 PRINT "FORMULA:(1=KNOWN,0=UNKNOWN)";
40 INPUT S
49 REM - IF FUNCTION IS KNOWN ENTER AT LINE 50 (DEFFNC(X)="FUNCTION")
50 DEFFNC(X)=X
60 PRINT "THE LOWER, UPPER LIMITS";
70 INPUT A,B
80 PRINT "INCREMENT OF X";
90 INPUT X1
98 REM - INCREMENT MUST DIVIDE INTERVAL INTO EQUAL SUBINTERVALS;
99 REM - IF NOT, CHANGE INCREMENT
100 T=(B-A)/X1:U=INT(T):IF T<>U THEN 80
110 IF S=1 THEN 150
119 REM - FORMULA NOT KNOWN; ENTER FUNCTION VALUE AT INTEGRATION LIMITS
120 PRINT "FIRST, LAST VALUE OF F(X)";
130 INPUT Y1,Y2
140 GOTO 170
149 REM - FORMULA KNOWN; CLACULATE F(X) AT INTEGRATION LIMITS
150 Y1=FNC(A)
160 Y2=FNC(B)
170 C=0
180 D=0
189 REM - LOOP FOR EACH INTERVAL
190 FOR I=1 TO (B-A)/X1-.5
200 IF S=1 THEN 240
209 REM - ENTER KNOWN FUNCTION VALUE AT EACH INTERVAL
210 PRINT "VALUE OF F(X) AT: "
211 PRINT TAB(10);"INTERVAL";I;
212 PRINT "(X=";A+I*X1;")";
220 INPUT Y
230 GOTO 250
239 REM - CALCULATE F(X) AT EACH SUBINTERVAL
240 Y=FNC(A+I*X1)
249 REM - INTERVAL EVEN OR ODD?
250 T2=I/2:R=INT(T2)
255 IF T2=R THEN 280
259 REM - SUM ALL ODD-INTERVAL FUNCTION VALUES
260 C=C+Y
```

```
270 GOTO 290
279 REM - SUM ALL EVEN-INTERVAL FUNCTION VALUES
280 D=D+Y
290 NEXT I
299 REM - COMPUTE INTEGRAL; PRINT
300 PRINT "INTEGRAL IS";
310 PRINT X1/3*(Y1+(C*4)+D*2+Y2)
320 END
```

Integration: Trapezoidal Rule

This program approximates the definite integral of a function. The integral is computed using the trapezoidal rule. You must provide the limits of integration and the number of intervals within the limits.

The function to be integrated must be entered before running the program. The function of x will be defined at line 30. For example, the function $f(x) = x^3$ will be entered as follows:

```
30 DEFFNC(X)=X↑3
```

Examples:

Find the definite integral of the function $f(x) = x^3$ between 0 and 2 with 10 and 20 intervals.

Find the definite integral of the function $f(x) = x^{-2}$ between 1 and 2 and 2 and 3 using 10 subintervals.

```
30 DEFFNC(X)=X↑3
RUN

INTEGRATION: TRAPEZOIDAL RULE

(ENTER 0,0 TO END PROGRAM)
INTEGRATION LIMITS (LOWER,UPPER)? 0,2
NUMBER OF INTERVALS? 10
INTEGRAL = 4.04000001

INTEGRATION LIMITS (LOWER,UPPER)? 0,2
NUMBER OF INTERVALS? 20
INTEGRAL = 4.01000001

INTEGRATION LIMITS (LOWER,UPPER)? 0,0

30 DEFFNC(X)=1/X↑2
RUN

INTEGRATION: TRAPEZOIDAL RULE

(ENTER 0,0 TO END PROGRAM)
INTEGRATION LIMITS (LOWER,UPPER)? 1,2
NUMBER OF INTERVALS? 10
INTEGRAL = .501455127

INTEGRATION LIMITS (LOWER,UPPER)? 2,3
NUMBER OF INTERVALS? 10
INTEGRAL = .177219009

INTEGRATION LIMITS (LOWER,UPPER)? 0,0
```

```
10 PRINT "□INTEGRATION: TRAPEZOIDAL RULE"
20 PRINT
29 REM - ENTER FUNCTION (DEFFNC(X)="FUNCTION")
30 PRINT "!ENTER FUNCTION AT LINE 30!!"
40 PRINT "(ENTER 0,0 TO END PROGRAM)"
50 PRINT "LOWER, UPPER LIMITS";
60 INPUT A,B
69 REM - END PROGRAM?
70 IF A=B THEN 190
80 PRINT "NUMBER OF INTERVALS";
90 INPUT N
100 I=0
109 REM - D IS THE SIZE OF EACH INTERVAL
110 D=(B-A)/N
119 REM - ADD UP THE AREA OF EACH TRAPEZOID
120 FOR J=A TO B+D STEP D
130 I=I+FNC(J)
140 NEXT J
149 REM - COMPUTE INTEGRAL, PRINT
150 I=(I-(FNC(A)+FNC(B))/2)*D
160 PRINT "INTEGRAL =";I
170 PRINT
179 REM - RESTART PROGRAM
180 GOTO 50
190 END
```

Integration: Gaussian Quadrature

This program approximates the definite integral of a function. You must provide the limits of integration and the number of intervals within the limits.

The interval of integration is divided into equal subintervals. The definite integral is computed over each subinterval using Gauss' formula. The integrals of the subintervals are summed to give the definite integral of the full interval.

You must enter the function to be integrated before running the program. The function of x will be defined at line 30. For example, the function $f(x) = x^3$ will be entered as follows:

```
30 DEFFNC(X)=X↑3
```

Examples:

Find the definite integral of the function $f(x) = x^3$ between 0 and 2 with 10 and 20 subintervals.

Find the definite integral of the function $f(x) = x^{-2}$ between 1 and 2 and 3 using 10 subintervals.

```
30 DEFFNC(X)=X↑3
RUN

INTEGRATION: GAUSSIAN QUADRATURE

LOWER, UPPER LIMITS? 0,2
NUMBER OF INTERVALS? 10
INTEGRAL = 4.00000004

CHANGE DATA AND RECOMPUTE?
(0=NO,1=LIMITS,2=INTERVALS)? 2
NUMBER OF INTERVALS? 20
INTEGRAL = 4.00000004

CHANGE DATA AND RECOMPUTE?
(0=NO,1=LIMITS,2=INTERVALS)? 0


30 DEFFNC(X)=1/X↑2
RUN

INTEGRATION: GAUSSIAN QUADRATURE

LOWER, UPPER LIMITS? 1,2
NUMBER OF INTERVALS? 10
INTEGRAL = .500000002

CHANGE DATA AND RECOMPUTE?
(0=NO,1=LIMITS,2=INTERVALS)? 1
LOWER, UPPER LIMITS? 2,3
INTEGRAL = .166666668
```

```
CHANGE DATA AND RECOMPUTE?
(0=NO,1=LIMITS,2=INTERVALS)? 0
```

```
10 PRINT "□INTEGRATION: GAUSSIAN QUADRATURE"
20 PRINT
29 REM - ENTER FUNCTION (DEFFNC(X)="FUNCTION")
30 PRINT "!ENTER FUNCTION AT LINE 30!!"
39 REM - ABSCISSAS AND WEIGHT FACTORS FOR 20-POINT GUASSIAN INTEGRATION
40 DATA .076526521,.15275339,.22778585
45 DATA .14917299,.37370609,.14209611
50 DATA .510867,.13168864,.63605368
55 DATA .11819453,.74633191,.10193012
60 DATA .83911697,.083276742,.91223443
65 DATA .062672048,.96397193,.04060143
70 DATA .9931286,.017614007
80 PRINT "LOWER, UPPER LIMITS";
90 INPUT X,Y
100 PRINT "NUMBER OF INTERVALS";
110 INPUT N
120 S=(Y-X)/N/2
130 T=X+S
140 R=0
149 REM - COMPUTE INTEGRAL FOR EACH SUBINTERVAL
150 FOR I=1 TO N
160 P=0
169 REM - COMPUTE SUMMATION FACTOR FOR EACH SUBINTERVAL
170 FOR J=1 TO 10
180 READ A,B
190 P=P+B*(FNC(S*A+T)+FNC(T-S*A))
200 NEXT J
210 RESTORE
220 R=R+P*S
230 T=T+2*S
240 NEXT I
250 PRINT "INTEGRAL =";R
260 PRINT
270 PRINT "CHANGE DATA AND RECOMPUTE?"
280 PRINT "(0=NO,1=LIMITS,2=INTERVALS)";
290 INPUT S
300 IF S=1 THEN 80
310 IF S=2 THEN 100
320 END
```

Derivative

This program calculates the derivative of a given function at a given point.

You must enter the function being evaluated before you run the program. The function will be entered in a definition statement at line 30. For example, to evaluate the equation $f(x) = x^2 + \cos(x)$ you would enter the following:

```
30 DEFFNC(X)=X↑2+COS(X)
```

Example:

Calculate the derivative of the equation $x^2 + \cos(x) = 0$ when $x = -1$, $x = 0$, and $x = 1$.

```
30 DEFFNC(X)=X↑2+COS(X)
RUN

DERIVATIVE

(ENTER X=99999 TO END)
DERIVATIVE AT X=? -1
                  IS-1.15852881
DERIVATIVE AT X=? 0
                  IS 2.43075192E-07
DERIVATIVE AT X=? 1
                  IS 1.15852833
DERIVATIVE AT X=? 99999
```

```
10 PRINT "□DERIVATIVE"
20 PRINT
29 REM - ENTER FUNCTION (DEFFNC(X)="FUNCTION")
30 PRINT "!ENTER FUNCTION AT LINE 30!"
40 PRINT "(ENTER X=99999 TO END)"
50 PRINT "DERIVATIVE AT X=";
60 INPUT X1
69 REM - TEST FOR END OF PROGRAM
70 IF X1=99999 THEN 160
80 D=0
89 REM - CALCULATE DIFFERENCE QUOTIENTS FOR POINTS APPROACHING X
90 FOR N=1 TO 10
100 D1=D
110 X=X1+.5↑N
115 Z=FNC(X)-FNC(X1)
120 D=Z/(X-X1)
130 NEXT N
139 REM - APPROXIMATE DERIVATIVE OF FUNCTION AT X, PRINT
140 PRINT "                    IS";2*D-D1
149 REM - RESTART PROGRAM
150 GOTO 50
160 END
```

Roots of Quadratic Equations

This program calculates the roots of a quadratic equation. The equation must be in the following form:

$$ax^2 + bx + c = 0$$

where *a,b,c* are real coefficients.

The formula used to calculate the roots is:

$$\text{root} = \frac{-b \pm \sqrt{b^2 - 4 \cdot a \cdot c}}{2 \cdot a}$$

Example:

Compute the roots of the following equations:

$$2x^2 + x - 1 = 0$$

$$x^2 + 4x + 6 = 0$$

```
ROOTS OF QUADRATIC EQUATIONS

COEFFICIENTS A,B,C? 2,1,-1
ROOTS (REAL):-1 , .5

MORE DATA (1=YES,0=NO)? 1

COEFFICIENTS A,B,C? 1,4,6
ROOTS (COMPLEX):-2  +OR-  1.41421356 I

MORE DATA (1=YES,0=NO)? 0
```

```
10 PRINT "□ROOTS OF QUADRATIC EQUATIONS"
20 PRINT
29 REM - ENTER COEFFICIENTS A,B,C OF A*X↑2 + B*X + C
30 PRINT "COEFFICIENTS A,B,C";
40 INPUT A,B,C
50 S=B↑2-4*A*C
60 R=SQR(ABS(S))
69 REM - COMPLEX ROOTS?
70 IF S<0 THEN 100
79 REM - CALCULATE ROOTS, LABEL, PRINT
80 PRINT "ROOTS (REAL):";(-B-R)/(2*A);
85 PRINT ",";(-B+R)/(2*A)
90 GOTO 110
100 PRINT "ROOTS (COMPLEX):";-B/(2*A);
105 PRINT "+OR-";R/(2*A);"I"
110 PRINT
119 REM - RESTART OR END PROGRAM?
```

```
120 PRINT "MORE DATA (1=YES,0=NO)";
130 INPUT X
140 IF X=1 THEN 20
150 END
```

Real Roots of Polynomials: Newton

This program calculates real roots of a polynomial with real coefficients. You must give an estimate of each root.

The calculations are performed using Newton's method for approximating roots of equations. The value of the error and derivative are included for each root calculated.

The equation you enter is presently limited to a degree of 10. You may enter a larger degree of equation by altering statements 30 and 40 of the program according to the following scheme:

```
30 DIM A(N+1),B(N+1)
40 FOR I=1 TO N+1
```

where N = degree of equation.

Example:

Find the roots of $4x^4 - 2.5x^2 - x + 0.5$

```
REAL ROOTS OF POLYNOMIALS

NEWTONIAN METHOD

DEGREE OF EQUATION? 4

COEFFICIENT A( 0 )? .5
COEFFICIENT A( 1 )? -1
COEFFICIENT A( 2 )? -2.5
COEFFICIENT A( 3 )? 0
COEFFICIENT A( 4 )? 4

GUESS? -.8

ROOT          ERROR             DERIVATIVE
.30357634     -2.91038305E-11   -2.070247

NEW VALUE (1=YES, 0=NO)? 0
NEW FUNCTION (1=YES, 0=NO)? 0
```

```
10 PRINT "REAL ROOTS OF POLYNOMIALS"
12 PRINT
15 PRINT "NEWTONIAN METHOD"
20 PRINT
28 REM - LIMIT A() AND B() TO N+1; WHEN THIS IS DONE, LOOP AT LINE 40
29 REM - SHOULD BE SET TO TEST FROM 1 TO N+1
30 DIM A(11),B(11)
39 REM - INITIALIZE ARRAY VARIABLES
40 FOR I=1 TO 11
50 A(I)=0
60 B(I)=0
70 NEXT I
```

```
80 PRINT "DEGREE OF EQUATION";
90 INPUT N
95 PRINT
100 FOR I=1 TO N+1
109 REM - ENTER COEFFICIENTS IN ORDER OF LESSER TO HIGHER DEGREE
110 PRINT "COEFFICIENT A(";I-1;")";
120 INPUT A(I)
130 NEXT I
140 FOR I=1 TO 10
149 REM - CALCULATE COEFFICIENT OF DERIVATIVE OF POLYNOMIAL
150 B(I)=A(I+1)*I
160 NEXT I
170 PRINT
179 REM - INITIALIZE GUESS
180 PRINT "GUESS";
190 INPUT X
200 Q=0
210 S=1
220 F1=0
230 F0=0
239 REM - COUNT ITERATIONS
240 Q=Q+1
250 FOR I=1 TO N+1
259 REM - CALCULATE VALUE OF FUNCTION
260 F0=F0+A(I)*S
269 REM - CALCULATE VALUE OF DERIVATIVE
270 F1=F1+B(I)*S
280 S=S*X
290 NEXT I
299 REM - TEST FOR A ZERO DERIVATIVE; IF YES, STOP SEARCH, PRINT
300 IF F1=0 THEN 360
309 REM - GET NEW GUESS USING PREVIOUS GUESS
310 S=X-F0/F1
319 REM - IF NEW GUESS EQUALS LAST GUESS THEN STOP SEARCH, PRINT
320 IF X=S THEN 380
329 REM - SAVE LAST GUESS
330 X=S
340 IF Q>100 THEN 490
350 GOTO 210
360 PRINT "DERIVATIVE = 0 AT X =";X
370 GOTO 180
380 PRINT
390 PRINT " ROOT";TAB(13);"ERROR";
395 PRINT TAB(29);"DERIVATIVE"
400 PRINT X;TAB(13);F0;TAB(29);F1
410 PRINT
419 REM - RERUN TO FIND ANOTHER ROOT IN SAME FUNCTION?
420 PRINT "NEW VALUE (1=YES, 0=NO)";
430 INPUT A
440 IF A=1 THEN 170
449 REM - RESTART OR END PROGRAM?
450 PRINT "NEW FUNCTION (1=YES, 0=NO)";
460 INPUT A
470 IF A=1 THEN 40
480 GOTO 550
489 REM - PRINT CALCULATED VALUES AFTER 100 ITERATIONS; SEARCH 100 MORE?
```

```
490 PRINT "100 ITERATIONS COMPLETED:"
500 PRINT "X =";X;"F(X) =";F0
510 PRINT "  CONTINUE (1=YES,0=NO)";
520 INPUT A
530 IF A=1 THEN 200
540 GOTO 420
550 END
```

Roots of Polynomials: Half-interval Search

This program calculates roots of polynomials within a given interval. The program first conducts a random search within the given interval for two points with opposite signs. If a change of sign is found, then the root is calculated by the half-interval search method. If there is no change of sign found, another interval will be asked for.

Errors may result in this program for a couple of reasons. First, a root may be calculated when it should not be. This may happen if the lowest point is so close to zero that a root is found due to round-off error. Second, two roots may be so close together that the program never finds the opposite signs between them. The result in this case is that neither root is calculated.

It is necessary to enter the equation before you run the program. The equation will be defined as a function of x at statement 30. For example, if you want to find roots of the function $f(x) = 4x^4-2.5x^2-x+.5$, you will enter:

```
30 DEFFNR(X)=4*X↑4-2.5*X↑2-X+.5
```

Example:

Find a root of the function $f(x) = 4x^4-2.5x^2-x+.5$.

```
30 DEFFNR(X)=4*X↑4-2.5*X↑2-X+.5
RUN


ROOTS OF POLYNOMIALS:
        HALF-INTERVAL SEARCH

(TO END SEARCH ENTER 0,0)

INTERVAL (LOWER,UPPER)? -1,0
NO CHANGE OF SIGN FOUND
INTERVAL (LOWER,UPPER)? 0,1
ROOT = .303574787

INTERVAL (LOWER,UPPER)? 0,0
```

```
10 PRINT"□ROOTS OF POLYNOMIALS:
15 PRINT"        HALF-INTERVAL SEARCH
20 PRINT
29 REM - ENTER FUNCTION (DEFFNR(X)="FUNCTION")
30 PRINT "!ENTER FUNTION AT LINE 30!"
40 DIM D(3)
50 PRINT "(TO END SEARCH ENTER 0,0)"
55 PRINT
59 REM - ESTABLISH INTERVAL OF RANDOM SEARCH
60 PRINT "INTERVAL (LOWER,UPPER)";
70 INPUT A,B
```

```
79 REM - TEST FOR USUABLE LIMITS ENTERED
80 IF A<>B THEN 120
89 REM - END PROGRAM?
90 IF A=0 THEN 430
100 PRINT "--INTERVAL LIMITS CANNOT";
105 PRINT "BE EQUAL--"
110 GOTO 60
120 IF A<B THEN 150
130 PRINT "--LOWER LIMIT MUST BE";
135 PRINT "ENTERED FIRST--"
140 GOTO 60
150 A1=SGN(FNR(A))
160 B1=SGN(FNR(B))
169 REM - TEST FOR ROOT AT EITHER LIMIT
170 IF A1*B1=0 THEN 360
179 REM - TEST FOR OPPOSITE SIGNS AT INTERVAL LIMITS
180 IF A1*B1<0 THEN 280
189 REM - LOOP TO SEARCH 1000 NUMBERS FOR OPPOSITE SIGNS IN FUNCTION
190 FOR I=1 TO 1000
200 X=A+RND(2)*(B-A)
210 X1=SGN(FNR(X))
219 REM - TEST FOR ROOT AT RANDOM NUMBER; IF YES, END SEARCH, PRINT
220 IF X1=0 THEN 400
229 REM - TEST FOR OPPOSITE SIGNS AT RANDOM NUMBER AND LOWER LIMIT
230 IF A1*X1<0 THEN 270
239 REM - TRY ANOTHER RANDOM NUMBER
240 NEXT I
250 PRINT "NO CHANGE OF SIGN FOUND"
260 GOTO 60
269 REM - CHANGE OF SIGN FOUND; CALCULATE ROOT
270 B=X
278 REM - STORE POSITIVE POINT IN D(3), NEGATIVE POINT IN D(1)
279 REM - D(1) AND D(3) BECOME INTERVAL LIMITS
280 D(2+A1)=A
290 D(2-A1)=B
299 REM - CALCULATE MIDPOINT BETWEEN THE TWO LIMITS
300 Y=(D(1)+D(3))/2
310 Y1=SGN(FNR(Y))
319 REM - TEST FOR ROOT AT MIDPOINT
320 IF Y1=0 THEN 400
329 REM - GET A NEW LIMIT TO CLOSE IN ON ROOT
330 D(2+Y1)=Y
339 REM - TEST FOR A VALUE CLOSE ENOUGH TO ZERO TO ASSUME A ROOT
340 IF ABS(D(1)-D(3))/ABS(D(1)                    +ABS(D(3)))<5E-6 THEN 400
349 REM - RETEST WITH NEW LIMITS
350 GOTO 300
359 REM - ROOT AT AN INTERVAL LIMIT; FIND WHICH LIMIT, PRINT
360 IF A1=0 THEN 390
370 Y=B
380 GOTO 400
390 Y=A
400 PRINT "ROOT =";Y
410 PRINT
419 REM - RESTART PROGRAM
420 GOTO 60
430 END
```

Trig Polynomial

This program solves a trigonometric function for a given angle. The function must be in the following form:

$$f(x) = A_1 \sin(x) + B_1 \cos(x) + A_2 \sin(2x) + B_2 \sin(2x) \ldots + A_n \sin(n \cdot x) + B_n \cos(n \cdot x)$$

where n = the number of pairs of coefficients.

The coefficients of the function are to be entered in a data statement at line 30. The data statement will include the number of pairs of coefficients (n) and the coefficients of the polynomial. It will be entered as follows:

$$30 \text{ DATA } n, A_1, B_1, A_2, B_2, \ldots A_n, B_n$$

Example:

Solve the following equation when the angle equals 45°, 90° and 105°:

$$f(x) = \sin(x) + 2 \cdot \cos(x) - 2 \cdot \sin(2x) + \cos(2x) + 5 \cdot \sin(3x) - 3 \cdot \cos(3x)$$

```
30 DATA 3,1,2,-2,1,5,-3
RUN


TRIG POLYNOMIAL

(ENTER ANGLE=99999 TO END)

ANGLE? 45
F( 45 )= 3.09558755

ANGLE? 90
F( 90 )=-2.83168119

ANGLE? 105
F( 105 )=-1.54684808

ANGLE? 99999
```

```
10 PRINT "□TRIG POLYNOMIAL"
20 PRINT
29 REM - ENTER NUMBER OF PAIRS OF TERMS AND COEFFICIENTS WITH DATA
   STATEMENT
30 PRINT "ENTER NUMBER OF PAIRS OF TERMS AND COEFFICIENTSS AT LINE 30!"
   :PRINT
40 PRINT "(ENTER ANGLE=99999 TO END)":PRINT
50 PRINT "ANGLE";
60 INPUT R
69 REM - END PROGRAM?
70 IF R=99999 THEN 180
```

```
79 REM - GET NUMBER OF PAIRS OF TERMS IN POLYNOMIAL
80 READ N
89 REM - LOOP TO GET VALUES OF COEFFICIENTS FROM DATA TABLE
90 FOR I=1 TO N
100 READ A,B
109 REM - CALCULATE VALUE OF FUNCTION AT ANGLE X
110 Z=Z+A*SIN(I*R)+B*COS(I*R)
120 NEXT I
129 REM - PRINT RESULTS
130 PRINT "F(";R;")=";Z
139 REM - PREPARE TO REREAD FUNCTION COEFFICIENTS
140 RESTORE
150 PRINT
160 Z=0
169 REM - RESTART PROGRAM
170 GOTO 50
180 END
```

Simultaneous Equations

This program solves a system of linear equations. The number of unknown coefficients in each equation must equal the number of equations being solved. You must enter the coefficients of each equation.

The dimension statement at line 30 limits the number of equations which may be solved. You may change this limit according to the following scheme:

```
30 DIM A(R,R+1)
```

where R = the maximum number of equations.

Example:

Solve the following system of equations:

$$\begin{aligned} x_1 + 2x_2 + 3x_3 &= 4 \\ 3x_1 + 6x_2 &= 1 \\ -3x_1 + 4x_2 - 2x_3 &= 0 \end{aligned}$$

```
30 DIM A(3,4)
RUN

SIMULTANEOUS EQUATIONS

NUMBER OF EQUATIONS? 3
COEFFICIENT MATRIX:

EQUATION 1
     COEFFICIENT 1 ? 1
     COEFFICIENT 2 ? 2
     COEFFICIENT 3 ? 3
     CONSTANT? 4

EQUATION 2
     COEFFICIENT 1 ? 3
     COEFFICIENT 2 ? 6
     COEFFICIENT 3 ? 0
     CONSTANT? 1

EQUATION 3
     COEFFICIENT 1 ? -3
     COEFFICIENT 2 ? 4
     COEFFICIENT 3 ? -2
     CONSTANT? 0

X 1 =-.356
X 2 = .344
X 3 = 1.222
```

```
10 PRINT "□SIMULTANEOUS EQUATIONS"
20 PRINT
29 REM - LIMIT A() TO A(R+1) WHERE R=MAXIMUM NUMBER OF EQUATIONS
30 DIM A(15,15)
40 PRINT "NUMBER OF EQUATIONS";
50 INPUT R
60 PRINT "COEFFICIENT MATRIX:"
70 FOR J=1 TO R
80 PRINT
85 PRINT "EQUATION";J
90 FOR I=1 TO R+1
100 IF I=R+1 THEN 130
110 PRINT "    COEFFICIENT";I;
120 GOTO 140
130 PRINT "CONSTANT";
140 INPUT A(J,I)
150 NEXT I
160 NEXT J
170 FOR J=1 TO R
178 REM - STATEMENTS 180 TO 220 FIND THE FIRST EQUATION WITH
179 REM - A NON-ZERO COEFFICIENT FOR THE CURRENT COLUMN
180 FOR I=J TO R
190 IF A(I,J)<>0 THEN 230
200 NEXT I
210 PRINT "NO UNIQUE SOLUTION"
220 GOTO 440
229 REM - STATEMENTS 230 TO 270 MOVE THAT EQUATION UP TO THE CURRENT
    ROW
230 FOR K=1 TO R+1
240 X=A(J,K)
250 A(J,K)=A(I,K)
260 A(I,K)=X
270 NEXT K
278 REM - STATEMENTS 280 TO 310 GENERATE A VALUE OF ONE IN THE FIRST
    NON-ZERO
279 REM - COLUMN OF THE CURRENT ROW
280 Y=1/A(J,J)
290 FOR K=1 TO R+1
300 A(J,K)=Y*A(J,K)
310 NEXT K
318 REM - STATEMENTS 320 TO 380 SUBTRACT CURRENT EQUATION FROM THE OT
    HER ROWS
320 FOR I=1 TO R
330 IFI=J THEN 380
340 Y=-A(I,J)
350 FOR K=1 TO R+1
360 A(I,K)=A(I,K)+Y*A(J,K)
370 NEXT K
380 NEXT I
389 REM - THIS PROCESS IS REPEATED FOR ALL EQUATIONS
390 NEXT J
400 PRINT
409 REM - PRINT SOLUTIONS
410 FOR I=1 TO R
420 PRINT "X";I;"=";INT(A(I,R+1)*1000+.5)/1000
430 NEXT I
440 END
```

Linear Programming

Courtesy: Harold Hanes
Earlham College
Richmond, Indiana

This program uses the simplex method to solve a linear programming problem. You must provide the coefficients of the objective function and the coefficients, relation and constant of each constraint. This information is entered in DATA statements before you run the program.

After you load the program, enter the DATA statements according to the following instructions. If you run more than one problem, remember to clear out all DATA statements from the previous problem before running the new problem. Our DATA statements occur at lines 30 through 35.

1) Arrange your problem constraints according to their relation, so that the "less than" inequalities precede the equalities, which in turn precede the "greater than" inequalities.
2) Type in as DATA the coefficients of the constraints, in the order the constraints were arranged in step 1. Do not include coefficients for slack, surplus, or artificial variables. Do include a '0' coefficient for any variable that doesn't appear in a particular constraint.
3) Type in as DATA the constants of the constraints (right-hand sides of the constraints) in the same order as you entered the rows of coefficients. These values cannot be negative.
4) Type in as DATA the coefficients of the objective function.

You must select whether the problem solution is to be a minimum or maximum value. The program also asks you to enter the total number of constraints and the number of variables to allow for each, and the number of "less than", "equal" and "greater than" constraints you are considering.

The dimension statement at line 180 limits the number of variables and constraints you may enter. You can change these limits according to the following scheme:

```
180 DIM A(C+2, V+C+G+1), B(C+2)
```

where: C = number of constraints
V = number of variables
G = number of "greater than" constraints

Example:

A manufacturer wishes to produce 100 pounds of an alloy which is 83% lead, 14% iron, and 3% antimony. He has available five alloys with the following compositions and prices:

	alloy 1	alloy 2	alloy 3	alloy 4	alloy 5
lead	90	80	95	70	30
iron	5	5	2	30	70
antimony	5	15	3	0	0
price	$6.13	$7.12	$5.85	$4.57	$3.96

How should he combine these alloys to get the desired product at minimum cost?

Note that this problem results in the following system of equations:

$$
\begin{array}{rcl}
x_1 + x_2 + x_3 + x_4 + x_5 & = & 100 \\
.90x_1 + .80x_2 + .95x_3 + .70x_4 + .30x_5 & = & 83 \\
.05x_1 + .05x_2 + .02x_3 + .30x_4 + .70x_5 & = & 14 \\
.05x_1 + .15x_2 + .03x_3 & = & 3 \\
6.13x_1 + 7.12x_2 + 5.85x_3 + 4.57x_4 + 3.96x_5 & = & Z\text{ (min)}
\end{array}
$$

```
LINEAR PROGRAMMING - SIMPLEX METHOD

1 MAXIMIZES -1 MINIMIZES? -1
# OF CONSTRAINTS,# OF VARIABLES? 4,5
# OF <,=,> CONSTRAINTS? 0,4,0


YOUR VARIABLES  1 THROUGH 5
ARTIFICIAL VARIABLES 6 THROUGH 9



ANSWERS:
PRIMAL VARIABLES:
VARIABLES VALUE
 2          10.4347826
 3          47.826087
 4          41.7391304
VALUE OF OBJECTIVE FUNCTION = 544.826087
```

```
1 REM - LINEAR PROGRAMMING
20 REM - *** DO THE FOLLOWING STEPS BEFORE RUNNING THE PROGRAM ***
21 REM - TYPE IN COEFFICIENTS OF '<','=','>' CONSTRAINTS IN DATA
   STATEMENTS;
22 REM - STARTING AT LINE 30, A SEPERATE DATA STATEMENT FOR EACH
   CONSTRAINT;
23 REM - (LINES 30-35 IN OUR EXAMPLE)
24 REM - TYPE IN CONSTANTS OF THE CONSTRAINTTS IN A DATA STATEMENT
   FOLLOWING;
25 REM - THE COEFFICIENT DATA, AND IN THE SAME ORDER AS THE CONSTRAINT
   DATA;
26 REM - WERE ENTERED (LINE 34 IN OUR EXAMPLE)
27 REM - TYPE IN COEFFICIENTS OF THE OBJECTIVE FUNCTION IN A DATA
   STATEMENT
28 REM - (LINE 35 IN OUR EXAMPLE) FOLLOWING THE CONSTANTS DATA
30 DATA 1,1,1,1,1
31 DATA .9,.8,.95,.7,.3
32 DATA .05,.05,.02,.3,.7
33 DATA .05,.15,.03,0,0
34 DATA 100,83,14,3
35 DATA 6.13,7.12,5.85,4.57,3.96
100 PRINT "]"
170 PRINT "LINEAR PROGRAMMING - ";
175 PRINT "SIMPLEX METHOD"
180 DIM A(6,10),B(6)
200 PRINT
210 PRINT "1 MAXIMIZES -1 MINIMIZES";
220 INPUT Z
230 Z=-Z
240 PRINT "# OF CONSTRAINTS,# OF ";
245 PRINT "VARIABLES";
250 INPUT M,N
260 PRINT "# OF <,=,> CONSTRAINTS";
270 INPUT L,E,G
280 IF M=L+E+G THEN 320
290 PRINT "INCONSISTENT DATA - ";
```

coeff of constraints

RHS

objective function

```
295 PRINT "TRY AGAIN"
300 GOTO 260
319 REM - THIS IS THE INITIALIZATION ROUTINE
320 C=N+M+G
330 C1=C+1
340 C2=N+L+G
350 M1=M+1
360 M2=M+2
380 PRINT
390 FOR I=1 TO M2
400 FOR J=1 TO C1
410 A(I,J)=0
420 NEXT J
430 NEXT I
440 FOR I=1 TO M
450 B(I)=0
460 NEXT I
470 FOR I=1 TO M
480 FOR J=1 TO N
490 READ A(I,J)
500 IF I<=L THEN 520
510 A(M1,J)=A(M1,J)-A(I,J)
520 NEXT J
530 IF I>L THEN 570
540 B(I)=N+I
550 A(I,N+I)=1
560 GOTO 630
570 B(I)=N+G+I
580 A(I,N+G+I)=1
590 IF I>L+E THEN 610
600 GOTO 630
610 A(I,N+I-E)=-1
620 A(M1,N+I-E)=1
630 NEXT I
640 FOR I=1 TO M
650 READ A(I,C1)
660 NEXT I
670 FOR J=1 TO N
680 READ A(M2,J)
690 A(M2,J)=Z*A(M2,J)
700 NEXT J
710 PRINT
730 PRINT "YOUR VARIABLES ";
731 PRINT 1; "THROUGH";N
740 IF L=0 THEN 760
750 PRINT "SLACK VARIABLES";
751 PRINT N+1; "THROUGH";N+L
760 IF G=0 THEN 780
770 PRINT "SURPLUS VARIABLES";
771 PRINT N+L+1; "THROUGH";C2
780 IF L=M THEN 970
790 PRINT "ARTIFICIAL VARIABLES";
791 PRINT C2+1; "THROUGH";C
800 M3=M1
810 GOSUB 1240
820 PRINT
830 FOR I1=1 TO M
```

```
840 IF B(I1)<=C2 THEN 950
850 IF A(I1,C1)<=.00001 THEN 880
860 PRINT "NO FEASIBLE SOLUTION"
870 GOTO 1700
880 FOR J1=1 TO C2
890 IF ABS(A(I1,J1))<=.00001 THEN 940
900 R=I1
910 S=J1
920 GOSUB 1490
930 J1=C2
940 NEXT J1
950 NEXT I1
970 PRINT
980 M3=M2
990 GOSUB 1240
1020 PRINT
1030 PRINT "ANSWERS:"
1040 PRINT "PRIMAL VARIABLES:"
1050 PRINT "VARIABLES","VALUE"
1060 FOR J=1 TO C2
1070 FOR I=1 TO M
1080 IF B(I)<>J THEN 1110
1090 PRINT J,A(I,C1)
1100 I=M
1110 NEXT I
1120 NEXT J
1130 IF L=0 THEN 1190
1140 PRINT "DUAL VARIABLES:"
1150 PRINT "VARIABLE","VALUE"
1160 FOR I=1 TO L
1170 PRINT I,-Z*A(M2,N+I)
1180 NEXT I
1190 PRINT "VALUE OF OBJECTIVE ";
1191 PRINT "FUNCTION =";-Z*A(M2,C1)
1200 PRINT
1210 PRINT
1220 PRINT
1230 GOTO 1700
1240 REM - OPTIMIZATION ROUTINE
1241 REM - FIRST PRICE OUT COLUMNS
1260 P=-.00001
1270 FOR J=1 TO C2
1280 IF A(M3,J)>=P THEN 1310
1290 S=J
1300 P=A(M3,J)
1310 NEXT J
1320 IF P=-.00001 THEN 1680
1330 GOSUB 1350
1340 GOSUB 1440
1345 GOTO 1260
1350 REM - NOW FIND WHICH VARIABLE LEAVE BASIS
1360 Q=1E38
1370 FOR I=1 TO M
1380 IF A(I,S)<=.00001 THEN 1420
1390 IF A(I,C1)/A(I,S)>=Q THEN 1420
1400 R=I
1410 Q=A(I,C1)/A(I,S)
```

```
1420 NEXT I
1430 RETURN
1440 IF Q=1E38 THEN 1470
1450 GOSUB 1490
1460 RETURN
1470 PRINT "THE SOLUTION IS UNBOUNDED"
1480 GOTO 1700
1490 REM - PERFORM PIVOTING
1500 P=A(R,S)
1510 FOR I=1 TO M2
1520 IF I=R THEN 1590
1530 FOR J=1 TO C1
1540 IF J=S THEN 1580
1550 A(I,J)=A(I,J)-A(I,S)*A(R,J)/P
1560 IF ABS(A(I,J))>=.00001 THEN 1580
1570 A(I,J)=0
1580 NEXT J
1590 NEXT I
1600 FOR J=1 TO C1
1610 A(R,J)=A(R,J)/P
1620 NEXT J
1630 FOR I=1 TO M2
1640 A(I,S)=0
1650 NEXT I
1660 A(R,S)=1
1670 B(R)=S
1680 RETURN
1700 END
```

Matrix Addition, Subtraction, Scalar Multiplication

This program adds or subtracts two matrices, or multiplies a matrix by a given scalar. You must input the value of each element of each matrix. To perform addition or subtraction the dimensions of the two matrices must be equal.

The dimension of the matrices may be increased or decreased depending on the amount of memory available in your system. Statement 30 may be changed to:

```
30 DIM A(X,Y), B(X,Y)
```

where (*X,Y*) is your limit on the dimension of the matrices.

Example:

Find the sum of the following matrices, then multiply the resultant matrix by 3.

$$\begin{bmatrix} 1 & 0 & -1 \\ 5 & 8 & .5 \\ -1 & 2 & 0 \end{bmatrix} \qquad \begin{bmatrix} -5 & -1 & 2 \\ 6 & -.1 & 0 \\ 3 & 4 & -2 \end{bmatrix}$$

```
MATRIX ADDITION, SUBTRACTION,
SCALAR MULTIPLICATION

1=ADDITION
2=SUBTARCTION
3=MULTIPLICATION

WHICH OPERATION? 1
DIMENSION OF MATRIX (R,C)? 3,3
MATRIX 1:
ROW 1
VALUE COLUMN 1 ? 1
VALUE COLUMN 2 ? 0
VALUE COLUMN 3 ? -1
ROW 2
VALUE COLUMN 1 ? 5
VALUE COLUMN 2 ? 8
VALUE COLUMN 3 ? .5
ROW 3
VALUE COLUMN 1 ? -1
VALUE COLUMN 2 ? 2
VALUE COLUMN 3 ? 0
MATRIX 2:
ROW 1
VALUE COLUMN 1 ? -5
VALUE COLUMN 2 ? -1
VALUE COLUMN 3 ? 2
ROW 2
VALUE COLUMN 1 ? 6
```

```
VALUE COLUMN 2 ? -.1
VALUE COLUMN 3 ? 0
ROW 3
VALUE COLUMN 1 ? 3
VALUE COLUMN 2 ? 4
VALUE COLUMN 3 ? -2
-4    -1     1
 11     7.9     .5
 2     6    -2

MORE DATA? (1=YES,0=NO)? 1

1=ADDITION
2=SUBTARCTION
3=MULTIPLICATION

WHICH OPERATION? 3
VALUE OF SCALAR? 3
DIMENSION OF MATRIX (R,C)? 3,3
MATRIX 1:
ROW 1
VALUE COLUMN 1 ? -4
VALUE COLUMN 2 ? -1
VALUE COLUMN 3 ? 1
ROW 2
VALUE COLUMN 1 ? 11
VALUE COLUMN 2 ? 7.9
VALUE COLUMN 3 ? .5
ROW 3
VALUE COLUMN 1 ? 2
VALUE COLUMN 2 ? 6
VALUE COLUMN 3 ? -2
-12    -3     3
 33     23.7     1.5
 6     18    -6

MORE DATA? (1=YES,0=NO)? 0
```

```
20 PRINT "□MATRIX ADDITION, ";
21 PRINT "SUBTRACTION,"
22 PRINT "SCALAR MULTIPLICATION"
25 PRINT
29 REM - ARRAYS SHOULD BE SET TO DIMENSONS OF MATRICES
30 DIM A(3,3),B(3,3)
40 PRINT "1=ADDITION"
50 PRINT "2=SUBTRACTION"
60 PRINT "3=SCALAR MULTIPLICATION"
65 PRINT
69 REM - SELECT OPERATION BY ENTERING THE OPERATION NUMBER (1-3)
70 PRINT "WHICH OPERATION";
80 INPUT D
89 REM - TEST FOR ADDITION OR SUBTRACTION
90 IF D<>3 THEN 120
100 PRINT "VALUE OF SCALAR";
110 INPUT S
120 PRINT "DIMENSION OF MATRIX (R,C)";
```

```
130 INPUT R,C
138 REM - LOOP TO ENTER MATRIX VALUES
139 REM - FOR SUBTRACTION, MATRIX 2 SUBTRACTED FROM MATRIX 1
140 FOR K=1 TO 2
150 IF K=2 THEN 180
160 PRINT "MATRIX 1:"
170 GOTO 190
180 PRINT "MATRIX 2:"
190 FOR J=1 TO R
200 PRINT "ROW";J
210 FOR I=1 TO C
220 PRINT "VALUE COLUMN";I;
230 IF K=2 THEN 260
240 INPUT A(J,I)
250 GOTO 270
260 INPUT B(J,I)
270 NEXT I
280 NEXT J
289 REM - ONLY ONE MATRIX USED FOR SCALAR MULTIPLICATION
290 IF D=3 THEN 310
300 NEXT K
308 REM - STATEMENTS 310 TO 410 PERFORM REQUESTED OPERATION AND PRINT
309 REM - RESULTANT MATRIX
310 FOR J=1 TO R
320 FOR I=1 TO C
330 IF D<>2 THEN 350
340 B(J,I)=-B(J,I)
350 IF D=3 THEN 380
360 PRINT A(J,I)+B(J,I);"  ";
370 GOTO 390
380 PRINT A(J,I)*S;"  ";
390 NEXT I
399 REM - ADVANCE OUTPUT DEVICE TO PRINT NEXT ROW
400 PRINT
410 NEXT J
420 PRINT
429 REM - RESTART OR END PROGRAM? USER INPUT REQUIRED
430 PRINT "MORE DATA? (1=YES, 0=NO)";
440 INPUT D
450 IF D=1 THEN 30
460 END
```

Matrix Multiplication

This program multiplies two matrices. The first matrix is multiplied by the second. You must input the elements of each matrix.

In order for this operation to be performed the number of rows in the first matrix must equal the number of columns in the second matrix.

The dimensions of the matrices are presently limited to 20 x 20. This limit may be increased or decreased by altering line 30 according to the following scheme:

```
30 DIM A(X,Y), B(Z,X)
```

where: (X,Y) = dimension of matrix 1
(Z,X) = dimension of matrix 2

Example:

Multiply matrix 1 by matrix 2.

$$1\begin{pmatrix} 2 & -1 & 4 & 1 & 2 \\ 1 & 0 & 1 & 2 & -1 \\ 2 & 3 & -1 & 0 & -2 \end{pmatrix}$$

$$2\begin{pmatrix} -2 & -1 & 2 \\ 0 & 2 & 1 \\ -1 & 1 & 4 \\ 3 & 0 & -1 \\ 2 & 1 & 2 \end{pmatrix}$$

```
30 DIM A(3,5),B(5,3)
RUN

MATRIX MULTIPLICATION

MATRIX 1 DIMENSION (R,C)? 3,5
MATRIX 2 DIMENSION (R,C)? 5,3
MATRIX 1:
ROW 1
VALUE COLUMN 1 ? 2
VALUE COLUMN 2 ? -1
VALUE COLUMN 3 ? 4
VALUE COLUMN 4 ? 1
VALUE COLUMN 5 ? 2
ROW 2
VALUE COLUMN 1 ? 1
VALUE COLUMN 2 ? 0
VALUE COLUMN 3 ? 1
VALUE COLUMN 4 ? 2
VALUE COLUMN 5 ? -1
```

```
ROW 3
VALUE COLUMN 1 ? 2
VALUE COLUMN 2 ? 3
VALUE COLUMN 3 ? -1
VALUE COLUMN 4 ? 0
VALUE COLUMN 4 ? -2

MATRIX 2:
ROW 1
VALUE COLUMN 1 ? -2
VALUE COLUMN 2 ? -1
VALUE COLUMN 3 ? 2
ROW 2
VALUE COLUMN 1 ? 0
VALUE COLUMN 2 ? 2
VALUE COLUMN 3 ? 1
ROW 3
VALUE COLUMN 1 ? -1
VALUE COLUMN 2 ? 1
VALUE COLUMN 3 ? 4
ROW 4
VALUE COLUMN 1 ? 3
VALUE COLUMN 2 ? 0
VALUE COLUMN 3 ? -1
ROW 5
VALUE COLUMN 1 ? 2
VALUE COLUMN 2 ? 1
VALUE COLUMN 3 ? 2

-1     2     22
 1    -1      2
-7     1    -1
```

```
10 PRINT "□MATRIX MULTIPLICATION"
20 PRINT
29 REM - ARRAYS A AND B SHOULD BE SET TO DIMENSIONS OF MATRICES
30 DIM A(20,20),B(20,20)
40 PRINT "MATRIX 1 DIMENSION (R,C)";
50 INPUT R1,C1
60 PRINT "MATRIX 2 DIMENSION (R,C)";
70 INPUT R2,C2
79 REM - NUMBER OF COLUMNS IN MATRIX 1 MUST EQUAL NUMBER OF ROWS IN
   MATRIX 2
80 IF C1=R2 THEN 110
90 PRINT "CANNOT BE MULTIPLIED"
100 GOTO 40
109 REM - ENTER MATIRX VALUES
110 PRINT "MATRIX 1:"
120 FOR J=1 TO R1
130 PRINT "ROW";J
140 FOR I=1 TO C1
150 PRINT "VALUE COLUMN";I;
160 INPUT A(J,I)
170 NEXT I
180 NEXT J
190 PRINT
```

```
200 PRINT "MATRIX 2:"
210 FOR J=1 TO R2
220 PRINT "ROW";J
230 FOR I=1 TO C2
240 PRINT "VALUE COLUMN";I;
250 INPUT B(J,I)
260 NEXT I
270 NEXT J
280 PRINT
289 REM - PERFORM MATRIX MULTIPLICATION, PRINT RESULTANT MATRIX
290 FOR I=1 TO R1
300 FOR J=1 TO C2
310 S=0
320 FOR K=1 TO C1
330 S=S+A(I,K)*B(K,J)
340 NEXT K
350 PRINT S;"  ";
360 NEXT J
369 REM - ADVANCE OUTPUT DEVICE TO PRINT NEXT ROW
370 PRINT
380 NEXT I
390 END
```

Matrix Inversion

This program inverts a square matrix. The inversion is performed by a modified Gauss-Jordan elimination method.

The dimensions of the matrices are presently limited to 20 × 20. This limit may be increased or decreased by altering line 30 according to the following scheme:

```
30 DIM A(R,R), B(R,R)
```

where R = number of rows (or columns) in the matrix.

Example:

Invert matrix A.

$$A \begin{Bmatrix} 3 & 5 & -1 & -4 \\ 1 & 4 & -.7 & -3 \\ 0 & -2 & 0 & 1 \\ -2 & 6 & 0 & .3 \end{Bmatrix}$$

```
MATRIX INVERSION

MATRIX DIMENSION? 4
MATRIX ELEMENTS:
ROW 1
VALUE COLUMN 1 ? 3
VALUE COLUMN 2 ? 5
VALUE COLUMN 3 ? -1
VALUE COLUMN 4 ? -4
ROW 2
VALUE COLUMN 1 ? 1
VALUE COLUMN 2 ? 4
VALUE COLUMN 3 ? -.7
VALUE COLUMN 4 ? -3
ROW 3
VALUE COLUMN 1 ? 0
VALUE COLUMN 2 ? -2
VALUE COLUMN 3 ? 0
VALUE COLUMN 4 ? 1
ROW 4
VALUE COLUMN 1 ? -2
VALUE COLUMN 2 ? 6
VALUE COLUMN 3 ? 0
VALUE COLUMN 4 ? .3

 .654       -.935     -.191      .014
 .198       -.283     -.103      .156
 .368      -1.955    -4.263     -.425
 .397       -.567      .793      .312
```

```
10 PRINT "□MATRIX INVERSION"
20 PRINT
29 REM - A() AND B() SHOULD BOTH BE SET TO THE DIMENSIONS OF THE MATIRX
30 DIM A(20,20),B(20,20)
39 REM - MATRIX IS SQUARE SO ONLY ONE DIMENSION IS NEEDED
40 PRINT "MATRIX DIMENSION";
50 INPUT R
60 PRINT "MATRIX ELEMENTS:"
69 REM - ENTER MATRIX ELEMENTS
70 FOR J=1 TO R
80 PRINT "ROW";J
90 FOR I=1 TO R
100 PRINT "VALUE COLUMN";I;
110 INPUT A(J,I)
120 NEXT I
130 B(J,J)=1
140 NEXT J
149 REM - STATEMENTS 150 TO 420 INVERT MATRIX
150 FOR J=1 TO R
160 FOR I=J TO R
170 IF A(I,J)<>0 THEN 210
180 NEXT I
190 PRINT "SINGULAR MATRIX"
200 GOTO 500
210 FOR K=1 TO R
220 S=A(J,K)
230 A(J,K)=A(I,K)
240 A(I,K)=S
250 S=B(J,K)
260 B(J,K)=B(I,K)
270 B(I,K)=S
280 NEXT K
290 T=1/A(J,J)
300 FOR K=1 TO R
310 A(J,K)=T*A(J,K)
320 B(J,K)=T*B(J,K)
330 NEXT K
340 FOR L=1 TO R
350 IF L=J THEN 410
360 T=-A(L,J)
370 FOR K=1 TO R
380 A(L,K)=A(L,K)+T*A(J,K)
390 B(L,K)=B(L,K)+T*B(J,K)
400 NEXT K
410 NEXT L
420 NEXT J
430 PRINT
439 REM - PRINT RESULTANT MATRIX
440 FOR I=1 TO R
450 FOR J=1 TO R
459 REM - ROUND OFF, PRINT
460 PRINT INT(B(I,J)*1000+.5)/1000,
461 PRINT " ";
470 NEXT J
479 REM - ADVANCE OUTPUT DEVICE TO PRINT NEXT LINE
480 PRINT
490 NEXT I
500 END
```

Permutations and Combinations

This program computes the number of permutations and combinations of *N* objects taken *D* at a time.

Examples:

How many permutations and combinations can be made of the 26 letters of the alphabet, taking five at a time?

How many different ways can 12 people sit on a bench if there is only room for two at a time?

```
PERMUTATIONS & COMBINATIONS

(ENTER 0 TO END PROGRAM)
TOTAL NUMBER OF OBJECTS? 26
SIZE OF SUBGROUP? 5
 7893600 PERMUTATIONS
 65780 COMBINATIONS

TOTAL NUMBER OF OBJECTS? 12
SIZE OF SUBGROUP? 2
 132 PERMUTATIONS
 66 COMBINATIONS

TOTAL NUMBER OF OBJECTS? 0
```

```
20 PRINT "□PERMUTATIONS & COMBINATIONS"
25 PRINT
30 PRINT "(ENTER 0 TO END PROGRAM)"
40 PRINT "TOTAL NUMBER OF OBJECTS";
50 INPUT N
59 REM - TEST FOR END OF PROGRAM
60 IF N=0 THEN 280
70 PRINT "SIZE OF SUBGROUP";
80 INPUT D
89 REM - SIZE OF SUBGROUP CANNOT BE LARGER THAN SIZE OF GROUP
90 IF D<=N THEN 130
100 PRINT "SUBGROUP TOO LARGE"
110 PRINT
120 GOTO 40
129 REM - LINES 130 TO 200 COMPUTE PERMUTATIONS
130 P=1
140 C=1
150 FOR I=N-D+1 TO N
159 REM - DON'T ALLOW NUMBER SIZE TO OVERFLOW MACHINE CAPACITY
160 IF 1.7E38/I>=P THEN 190
170 PRINT "> 1.7E38 PERMUTATIONS"
180 GOTO 280
190 P=P*I
200 NEXT I
```

```
209 REM - COMPUTE INTERMEDIATE FACTORIAL FOR COMBINATIONS
210 FOR J=2 TO D
220 C=C*J
230 NEXT J
240 PRINT P;"PERMUTATIONS"
250 PRINT P/C;"COMBINATIONS"
260 PRINT
269 REM - RESTART PROGRAM
270 GOTO 40
280 END
```

Mann-Whitney *U* Test

This program performs the Mann-Whitney *U* test on samples from two populations.

The dimension statement on line 30 limits the size of the samples. You can increase or decrease the dimension limits according to the following scheme:

```
30 DIM X(M), Y(N)
```

where: *M* = maximum size of first sample
N = maximum size of second sample

Example:

A group of ten women and a group of ten men were asked to rate the flavor of a frozen T.V. dinner on a scale of one to ten. The table below lists the scores. Count the number of times the women's scores are lower than the men's, and vice-versa.

women	1	3	4	3	6	8	9	7	8	4
men	7	9	8	5	10	9	10	6	5	2

```
30 DIM X(10),Y(10)
RUN
MANN-WHITNEY U-TEST

SAMPLE 1 :
  SIZE? 10
   DATA 1 ? 1
   DATA 2 ? 3
   DATA 3 ? 4
   DATA 4 ? 3
   DATA 5 ? 6
   DATA 6 ? 8
   DATA 7 ? 9
   DATA 8 ? 7
   DATA 9 ? 8
   DATA 10 ? 4

SAMPLE 2 :
  SIZE? 10
   DATA 1 ? 7
   DATA 2 ? 9
   DATA 3 ? 8
   DATA 4 ? 5
   DATA 5 ? 10
   DATA 6 ? 9
   DATA 7 ? 10
   DATA 8 ? 6
   DATA 9 ? 5
   DATA 10 ? 2

FIRST PRECEDING, U = 71.5
SECOND PRECEDING, U = 28.5
```

```
10 PRINT "□MANN-WHITNEY U-TEST"
20 PRINT
28 REM - SET MAXIMUM SAMPLE SIZE TO X(M),Y(N) (WHERE M=MAXIMUM SIZE OF
29 REM - SAMPLE 1, N=MAXIMUM SIZE OF SAMPLE 2)
30 DIM X(25),Y(25)
40 DIM N(2)
49 REM - INPUT THE TWO SAMPLES
50 FOR I=1 TO 2
60 PRINT "SAMPLE";I;":"
70 PRINT "  SIZE";
80 INPUT N(I)
90 FOR J=1 TO N(I)
100 PRINT "   DATA";J;
110 INPUT Y(J)
120 NEXT J
129 REM - SORT EACH SAMPLE
130 FOR J=1 TO N(I)
140 FOR K=1 TO N(I)-J
150 C=Y(K)
170 IF Y(K)<Y(K+1) THEN 200
180 Y(K)=Y(K+1)
190 Y(K+1)=C
200 NEXT K
210 NEXT J
220 PRINT
229 REM - TRANSFER FIRST EXAMPLE TO X-ARRAY
230 IF I=2 THEN 270
240 FOR J=1 TO N(1)
250 X(J)=Y(J)
260 NEXT J
270 NEXT I
279 REM - ADD UP RANKS
280 R=1
290 I=0
300 J=0
310 I=I+1
320 J=J+1
330 IF I>N(1) THEN 580
340 IF J>N(2) THEN 620
350 IF X(I)<Y(J) THEN 620
360 IF Y(J)<X(I) THEN 590
369 REM - LINES 370 TO 570 HANDLE EQUAL SCORES FROM BOTH SAMPLES
370 K=2
380 M=I
390 L=J
400 R1=2*R+1
410 R=R+2
420 I=I+1
430 J=J+1
440 IF I>N(1) THEN 480
450 IF X(I)<>(I-1) THEN 480
460 I=I+1
470 GOTO 510
480 IF J>N(2) THEN 550
490 IF Y(J)<>(J-1) THEN 550
500 J=J+1
510 R1=R1+R
```

```
520 R=R+1
530 K=K+1
540 GOTO 440
550 X=X+(I-M)*R1/K
560 Y=Y+(J-L)*R1/K
570 GOTO 330
580 IF J>N(2) THEN 660
590 Y=Y+R
600 J=J+1
610 GOTO 640
620 X=X+R
630 I=I+1
640 R=R+1
650 GOTO 330
659 REM - U1=NUMBER OF TIMES SAMPLE 1 SCORES PRECEDE SAMPLE 2 SCORES
660 U1=N(1)*N(2)+N(1)*(N(1)+1)/2-X
669 REM - U2=NUMBER OF TIMES SAMPLE 2 SCORES PRECEDE SAMPLE 1 SCORES
670 U2=N(1)*N(2)+N(2)*(N(2)+1)/2-Y
680 PRINT
690 PRINT "FIRST PRECEDING, U =";U1
700 PRINT "SECOND PRECEDING, U =";U2
710 END
```

Mean, Variance, Standard Deviation

This program calculates the arithmetic mean, variance and standard deviation of grouped or ungrouped data. The data may represent the entire population or just a sample.

Examples:

There are ten people in a hotel lobby, aged 87, 53, 35, 42, 9, 48, 51, 60, 39 and 44. What would the mean, variance and standard deviation of the ages of all the people in the hotel be using the people in the lobby as a sample?

Find the mean, variance and standard deviation of the ages of the cream cheese on a market shelf. The table below lists the age distribution of 50 packages. Assume the table shows the store's entire inventory. What if it is only a sample of the inventory?

age	1	2	3	4	5	6
quantity	15	10	9	6	7	3

cream cheese

```
MEAN, VARIANCE, STANDARD DEVIATION

METHOD (0=POPULATION, 1=SAMPLE)? 1
DATA (0=GROUPED, 1=UNGROUPED)? 1
NUMBER OF OBSERVATIONS? 10
ITEM 1 ? 87
ITEM 2 ? 53
ITEM 3 ? 35
ITEM 4 ? 42
ITEM 5 ? 9
ITEM 6 ? 48
ITEM 7 ? 51
ITEM 8 ? 60
ITEM 9 ? 39
ITEM 10 ? 44

MEAN          VARIANCE
 46.8          389.733332

STANDARD DEVIATION
 19.7416649

MORE DATA (1=YES, 0=NO)? 1

METHOD (0=POPULATION, 1=SAMPLE)? 0
DATA (0=GROUPED, 1=UNGROUPED)? 0
NUMBER OF OBSERVATIONS? 6
ITEM, FREQUENCY 1 ? 1,15
ITEM, FREQUENCY 2 ? 2,10
ITEM, FREQUENCY 3 ? 3,9
ITEM, FREQUENCY 4 ? 4,6
```

```
ITEM, FREQUENCY 5 ? 5,7
ITEM, FREQUENCY 6 ? 6,3

MEAN        VARIANCE
 2.78        2.57159999

STANDARD DEVIATION
 1.6036209

METHOD (0=POPULATION, 1=SAMPLE)? 1
DATA (0=GROUPED, 1=UNGROUPED)? 0
NUMBER OF OBSERVATIONS? 6
ITEM, FREQUENCY 1 ? 1,15
ITEM, FREQUENCY 2 ? 2,10
ITEM, FREQUENCY 3 ? 3,9
ITEM, FREQUENCY 4 ? 4,6
ITEM, FREQUENCY 5 ? 5,7
ITEM, FREQUENCY 6 ? 6,3

MEAN        VARIANCE
 2.78        2.62408162

STANDARD DEVIATION
 1.6199017

MORE DATA (1=YES, 0=NO)? 0
```

```
10 PRINT "MEAN, VARIANCE, ";
20 PRINT "STANDARD DEVIATION"
25 PRINT
30 PRINT "METHOD (0=POPULATION, ";
31 PRINT "1=SAMPLE)";
40 INPUT S
50 PRINT "DATA (0=GROUPED, ";
51 PRINT "1=UNGROUPED)";
60 INPUT K
70 PRINT "NUMBER OF OBSERVATIONS";
80 INPUT N
90 R=0
100 M=0
110 P=0
120 IF K=1 THEN 230
129 REM - FOR GROUPED DATA
130 FOR I=1 TO N
140 PRINT "ITEM, FREQUENCY";I;
150 INPUT A,B
159 REM - ACCUMULATE ENTERED VALUES
160 R=R+B*A
169 REM - ACCUMULATE INTERMEDIATE VALUES FOR VARIANCE
170 P=P+B
180 M=M+B*A↑2
190 NEXT I
199 REM - CALCULATE MEAN AND VARIANCE
200 R=R/P
210 V=(M-P*R↑2)/(P-S)
```

```
219 REM - PRINT RESULTS
220 GOTO 310
229 REM - FOR UNGROUPED DATA
230 FOR I=1 TO N
240 PRINT "ITEM";I;
250 INPUT D
259 REM - ACCUMUATE ENTERED VALUES
260 P=P+D
269 REM - ACCUMULATE INTERMEDIATE VALUES FOR VARIANCE
270 M=M+D↑2
280 NEXT I
289 REM - CALCULATE MEAN AND VARIANCE, PRINT
290 R=P/N
300 V=(M-N*R↑2)/(N-S)
310 PRINT
319 REM - PRINT RESULTS
320 PRINT "MEAN","VARIANCE"
330 PRINT R,V
331 PRINT
332 PRINT "STANDARD DEVIATION"
333 PRINT SQR(V)
340 PRINT
349 REM - RESTART OR END PROGRAM? USER INPUT REQUIRED
350 PRINT "MORE DATA (1=YES, 0=NO)";
360 INPUT S
370 IF S=1 THEN 25
380 END
```

Geometric Mean and Deviation

This program computes the geometric mean and standard deviation of a set of data.

Example:

Find the geometric mean and standard deviation of 3, 5, 8, 3, 7, 2.

```
GEOMETRIC MEAN AND DEVIATION

(TO END PROGRAM, ENTER 0)
NUMBER OF OBSERVATIONS? 6
ITEM 1 ? 3
ITEM 2 ? 5
ITEM 3 ? 8
ITEM 4 ? 3
ITEM 5 ? 7
ITEM 6 ? 2
GEOMETRIC MEAN = 4.14068084
GEOMETRIC DEVIATION = 1.72368956

NUMBER OF OBSERVATIONS? 0
```

```
10 PRINT "□GEOMETRIC MEAN AND DEVIATION"
20 PRINT
30 PRINT "(TO END PROGRAM, ENTER 0)"
40 PRINT "NUMBER OF OBSERVATIONS";
50 INPUT N
59 REM - TEST FOR END OF PROGRAM
60 IF N=0 THEN 200
69 REM - COMPUTE WHICH ROOT TO USE
70 P=1/N
80 M=1
90 FOR I=1 TO N
100 PRINT "ITEM";I;
110 INPUT D
119 REM - ITERATIVELY COMPUTE MEAN
120 M=M*D↑P
129 REM - ACCUMULATE INTERMEDIATE TERM FOR DEVIATION
130 Q=Q+LOG(D)↑2
140 NEXT I
149 REM - COMPUTE DEVIATION
150 R=EXP(SQR(Q/(N-1)-(N/(N-1)*(LOG(M))↑2)))
160 PRINT "GEOMETRIC MEAN =";M
170 PRINT "GEOMETRIC DEVIATION =";R
180 PRINT
189 REM - RESTART PROGRAM
190 GOTO 40
200 END
```

Binomial Distribution

This program calculates the probability of obtaining a given number of successes in a given number of Bernoulli trials. You must provide the probability of success on a single trial.

Examples:

What is the probability of getting three heads in five tosses of a fair coin?

What is the probability that in five rolls of a fair die, a one (1) appears twice?

```
BINOMIAL DISTRIBUTION

(TO END PROGRAM ENTER 0)
NUMBER OF TRIALS? 5
EXACT NUMBER OF SUCCESSES? 3
PROBABILITY OF SUCCESS? .5

PROBABILITY OF 3 SUCCESSES
IN 5 TRIALS = .3125

NUMBER OF TRIALS? 5
EXACT NUMBER OF SUCCESSES? 2
PROBABILITY OF SUCCESS? .166666667

PROBABILITY OF 2 SUCCESSES
IN 5 TRIALS = .160751029

NUMBER OF TRIALS? 0
```

```
10 PRINT "□BINOMIAL DISTRIBUTION"
20 PRINT
30 DIM M(3)
40 PRINT "(TO END PROGRAM ENTER 0)"
50 PRINT "NUMBER OF TRIALS";
60 INPUT N
70 IF N=0 THEN 270
80 PRINT "EXACT NUMBER OF SUCCESSES";
90 INPUT X
100 PRINT "PROBABILITY OF SUCCESS";
110 INPUT P
119 REM - COMPUTE THE FACTORIALS
120 M(1)=N
130 M(2)=X
140 M(3)=N-X
150 FOR I=1 TO 3
160 IF M(I)=0 THEN 220
170 A=1
180 FOR J=1 TO M(I)
190 A=A*J
```

```
200 NEXT J
210 M(I)=LOG(A)
220 NEXT I
229 REM - USING THE COMPUTED FACTORIALS, COMPUTE PROBABILITY
230 R=EXP(M(1)-M(2)-M(3)+X*LOG(P)+(N-X)*LOG(1-P))
235 PRINT
240 PRINT"PROBABILITY OF";X;
245 PRINT "SUCCESSES"
247 PRINT "IN";N;"TRIALS =";R
250 PRINT
259 REM - RESTART PROGRAM
260 GOTO 50
270 END
```

Poisson Distribution

Using the Poisson distribution this program calculates the probability of an event occurring a given number of times. You must know the expected frequency of the event.

Example:

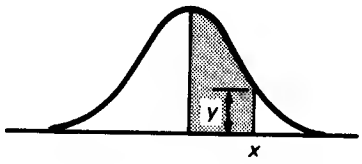
2000 people are injected with a serum. The probability of any one person having a bad reaction is .001. Thus we can expect two (.001•2000=2) individuals will suffer a bad reaction. What is the probability that four people will have bad reactions? Only one person?

```
POISSON DISTRIBUTION

(TO END PROGRAM ENTER 0)

CALCULATED FREQUENCY? 2
TEST FREQUENCY? 4
PROBABILITY OF 4
OCCURRENCES = .0902235222

CALCULATED FREQUENCY? 2
TEST FREQUENCY? 1
PROBABILITY OF 1
OCCURRENCES = .270670566

CALCULATED FREQUENCY? 0
```

```
10 PRINT "POISSON DISTRIBUTION"
15 PRINT
20 PRINT "(TO END PROGRAM ENTER 0)"
30 PRINT
40 PRINT "CALCULATED FREQUENCY";
50 INPUT L
59 REM - END PROGRAM?
60 IF L=0 THEN 180
70 PRINT "TEST FREQUENCY";
80 INPUT X
89 REM - COMPUTE FACTORIAL
90 A=1
100 FOR I=1 TO X
110 A=A*I
120 NEXT I
129 REM - COMPUTE PROBABILITY
130 A=LOG(A)
140 A=EXP(-L+X*LOG(L)-A)
150 PRINT "PROBABILITY OF";X
160 PRINT "OCCURRENCES =";A
169 REM - RESTART PROGRAM
170 GOTO 30
180 END
```

Normal Distribution

This program calculates the probability and frequency of given values on a standard normal distribution curve. You can use non-standard variables if you know the mean and standard deviation.



Standard normal distribution

The shaded area represents the probability of x. y corresponds to the frequency of x.

The normal probability is approximated using the following formula:

$$\text{probability} = 1 - r(a_1 t + a_2 t^2 + a_3 t^3) + \epsilon(x)$$

where:
$$a_1 = .4361836$$
$$a_2 = -.1201676$$
$$a_3 = .9372980$$
$$r = (e^{-x^2/2})(2\pi)^{-1/2}$$
$$t = (1 + .3326x)^{-1}$$
$$|\epsilon(x)| < 10^{-5}$$

Example:

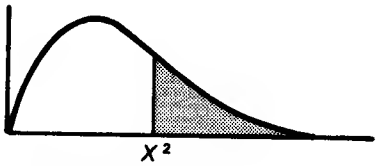
The mean weight of the male students at a college is 150 pounds. The standard deviation is 15 pounds. If the weights are normally distributed, what is the probability that a student weighs between 150 and 180 pounds? Between 130 and 150 pounds?

```
NORMAL DISTRIBUTION

(0=STANDARD, 1=NON-STANDARD)
WHICH TYPE OF VARIABLE? 1
MEAN? 150
STANDARD DEVIATION? 15

(TO END PROGRAM X=99999)
X =? 180
FREQUENCY = .0539909665
PROBABILITY = .977241178

X =? 130
FREQUENCY = .164010074
PROBABILITY = .908798028

X =? 99999
```

```
10 PRINT "□NORMAL DISTRIBUTION"
20 PRINT
30 PRINT "(0=STANDARD,1=NON-STANDARD)"
40 PRINT "WHICH TYPE OF VARIABLE";
50 INPUT S
60 IF S=0 THEN 120
69 REM - LINES 70 TO 110 REQUEST 'NON-STANDARD' VARIABLE DATA
70 PRINT "MEAN";
80 INPUT M
90 PRINT "STANDARD DEVIATION";
100 INPUT S
110 GOTO 130
120 S=1
130 PRINT
140 PRINT "(TO END PROGRAM X=99999)"
150 PRINT "X =";
160 INPUT X
170 IF X=99999 THEN 290
180 X=ABS((X-M)/S)
190 R=EXP(-(X↑2)/2)/2.5066282746
200 PRINT "FREQUENCY =";R
210 Z=X
220 T=1/(1+.33267*ABS(X))
230 T=1-R*(.4361836*T-.120167*T↑2+.937298*T↑3)
240 IF Z>=0 THEN 260
250 T=1-T
260 PRINT "PROBABILITY =";T
270 PRINT
279 REM - RESTART PROGRAM
280 GOTO 150
290 END
```

Chi-square Distribution

This program calculates the tail-end value for points on a chi-square (X^2) distribution curve. You must provide the value of X^2 and the degrees of freedom.



Chi-square distribution

The shaded area represents the tail-end value of X^2.

The X^2 distribution function is calculated using the following formulas:

$$\text{with } v \text{ odd, tail-end value} = 1 - \frac{(X^2)^{(v+1)/2} \cdot e^{-x^2/2}}{1 \cdot 3 \cdot 5 \ldots \cdot v} \cdot \left(\frac{2}{X^2 \pi}\right)^{1/2} \cdot Z$$

$$\text{with } v \text{ even, tail-end value} = 1 - \frac{(X^2)^{v/2} \cdot e^{-x^2/2}}{2 \cdot 4 \cdot \ldots \cdot v} \cdot Z$$

where: v = degrees of freedom

$$Z = 1 + \sum_{m=1}^{\infty} \frac{(X^2)^m}{(v+2) \cdot (v+4) \cdot \ldots (v+2m)}$$

Since the summation in the calculation of Z cannot actually extend to infinity, we stop summation when the next term is less than a chosen level of precision. The computational precision is limited to approximately 10^{-7}.

Example:

Of a group of 168 people who complained they did not sleep well, 54 were given sleeping pills and the remainder received placebos. They were later asked whether or not the pills had helped them sleep. The X^2 statistic for this study was computed to be 2.571108 with one degree of freedom. What is the tail-end value?

```
CHI-SQUARE DISTRIBUTION

(TO END PROGRAM ENTER 0)
DEGREES OF FREEDOM? 1
CHI-SQUARE? 2.571108
TAIL END VALUE = .108831483

DEGREES OF FREEDOM? 0
```

```
10 PRINT "□CHI-SQUARE DISTRIBUTION"
20 PRINT
30 PRINT "(TO END PROGRAM ENTER 0)"
40 PRINT "DEGREES OF FREEDOM";
50 INPUT V
60 IF V=0 THEN 280
70 PRINT "CHI-SQUARE";
80 INPUT W
89 REM - R=DENOMINATOR PRODUCT
90 R=1
100 FOR I=V TO 2 STEP -2
110 R=R*I
120 NEXT I
129 REM - K=THE NUMERATOR PRODUCT
130 K=W↑(INT((V+1)/2))*EXP(-W/2)/R
139 REM - THE PI FACTOR IS USED ONLY WHEN DEGREES OF FREEDOM ARE ODD
140 IF INT(V/2)=V/2 THEN 170
150 J=SQR(2/W/π)
160 GOTO 180
169 REM - L (SUMMATION FACTOR) CALCULATED LINES 170-240
170 J=1
180 L=1
190 M=1
200 V=V+2
210 M=M*W/V
219 REM - CHECK FOR END OF SUMMATION
220 IF M<.0000001 THEN 250
230 L=L+M
240 GOTO 200
250 PRINT "TAIL END VALUE =";1-J*K*L
260 PRINT
269 REM - RESTART PROGRAM
270 GOTO 40
280 END
```

OPTION

You may wish to compute the percentile rather than the tail-end value. This value corresponds to the unshaded area in the figure above. The program changes necessary are listed following the example below.

Example:

What is the percentile in the example above?

```
CHI-SQUARE DISTRIBUTION

(TO END PROGRAM ENTER 0)
DEGREES OF FREEDOM? 1
CHI-SQUARE? 2.571108
PERCENTILE= .891168517

DEGREES OF FREEDOM? 0
```

```
1 REM - OPTION 250
10 PRINT "□CHI-SQUARE DISTRIBUTION"
 .
 .
 .
240 GOTO 200
250 PRINT "PERCENTILE =";J*K*L
260 PRINT
269 REM - RESTART PROGRAM
270 GOTO 40
280 END
```

Chi-square Test

This program calculates the chi-square (X^2) statistic and degrees of freedom associated with a given contingency table. The expected value for each cell and X^2 contribution from each cell are also printed.

The dimension statement at line 30 limits the size of the contingency table. You can change the dimensions according to the following scheme:

```
30 DIM V1(R*C), V2(C), A(R)
```

where: R = number of rows in the contingency table
C = number of columns in the contingency table

Example:

Of a group of people who complained they could not sleep well, some were given sleeping pills while others were given placebos. Later they were asked whether or not the pills had helped them sleep. The results are detailed in the table below. What is the value of the X^2 statistic?

	slept well	slept poorly
sleeping pill	44	10
placebo	81	35

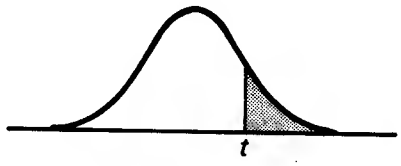
```
30 DIM V1(4),V(2),A(2)
RUN

CHI-SQUARE TEST

NUMBER OF ROWS? 2
NUMBER OF COLUMNS? 2
CONTINGENCY TABLE:
ROW 1
     ELEMENT 1 ? 44
     ELEMENT 2 ? 10
ROW 2
     ELEMENT 1 ? 81
     ELEMENT 2 ? 35

OBSERVED     EXPECTED CHI^2 CONTRIBUTION
 COLUMN 1
    44      39.7058824     .362549019
    81      85.2941177     .168772821
 COLUMN 2
    10      14.2941176     1.00708061
    35      30.7058824     .468813387

CHI-SQUARE = 2.00721584
DEGREES OF FREEDOM = 1
```

```
10 PRINT "□CHI SQUARE TEST"
20 PRINT
28 REM - LIMIT SIZE OF CONTINGENCY TABLES TO V1(R*C), V2(C), A(R);
29 REM - WHERE R=NUMBER OF ROWS, C=NUMBER OF COLUMNS
30 DIM V1(25),V2(2),A(2)
40 PRINT "NUMBER OF ROWS";
49 REM - INPUT CONTINGENCY TABLE LINES 50 TO 150
50 INPUT R
60 PRINT "NUMBER OF COLUMNS";
70 INPUT C
80 PRINT "CONTINGENCY TABLE:"
90 FOR I=1 TO R
100 PRINT "ROW";I
110 FOR J=1 TO C
120 PRINT "    ELEMENT";J;
130 INPUT V1((I-1)*C+J)
140 NEXT J
150 NEXT I
160 PRINT
169 REM - ADD UP MARGINAL FREQUENCIES FOR EACH ROW
170 L=0
180 M=1
190 FOR I=1 TO R
200 FOR J=1 TO C
210 A(I)=A(I)+V1(M)
220 M=M+1
230 NEXT J
240 L=L+A(I)
250 NEXT I
260 N=R*C
269 REM - ADD UP MARGINAL FREQUENCIES FOR EACH COLUMN
270 FOR I=1 TO C
280 FOR J=I TO N STEP C
290 V2(I)=V2(I)+V1(J)
300 NEXT J
310 NEXT I
320 Z=0
330 PRINT "OBSERVED ";"   EXPECTED ";
331 PRINT "CHI↑2 CONTRIBUTION"
340 FOR I=1 TO C
350 PRINT " COLUMN";I
360 FOR J=1 TO R
369 REM - P=EXPECTED CELL VALUE
370 P=A(J)*V2(I)/L
375 X=I+(J-1)*C
379 REM - USE YATES' CORRECTION FOR CONTINUITY IN 2 X 2 CHI-SQUARE TES'
380 IF R<>2 THEN 390
381 IF C<>2 THEN 390
382 Y=(ABS(V1(X)-P)-.5)↑2/P
383 GOTO 400
389 REM - Y=CHI-SQUARE CONTRIBUTION FROM THIS CELL
390 Y=(V1(X)-P)↑2/P
399 REM - Z=TOTAL CHI-SQUARE VALUE
400 Z=Z+Y
410 PRINT TAB(3);V1(X);TAB(10);P;
411 PRINT TAB(24);Y
420 NEXT J
```

```
430 NEXT I
440 PRINT
450 PRINT "CHI-SQUARE =";Z
460 PRINT "DEGREES OF FREEDOM =";
461 PRINT (C-1)*(R-1)
470 END
```

Student's t -distribution

This program calculates right-tail values for points on a t -distribution curve. You must provide the value of t and the degrees of freedom.



Student's t -distribution

The shaded area represents the right-tail value for t.

The right-tail value is approximated using the following formula:

$$\text{right-tail value} = \frac{1}{2}(1 + a_1 x + a_2 x^2 + a_3 x^3 + a_4 x^4)^{-4} + \epsilon(x)$$

where: $a_1 = .196854$
$a_2 = .115194$
$a_3 = .000344$
$a_4 = .019527$

$$x = t^{2/3}\left(1 - \frac{2}{9d}\right) - \frac{7}{9}\left(\frac{2}{9} + t^{4/8} \cdot \frac{2}{9d}\right)^{-1/2}$$

d = degrees of freedom

$$|\epsilon(x)| < 2.5 \cdot 10^{-4}$$

Examples:

What is the right-tail value when the t -value is 2.921 and there are 16 degrees of freedom?

What is the right-tail value when the t -value is 11.178 and there are 5 degrees of freedom?

```
STUDENT'S T-DISTRIBUTION

(TO END PROGRAM ENTER 0)
T VALUE? 2.921
DEGREES OF FREEDOM? 16
RIGHT TAIL VALUE = 9.7E-04

T VALUE? 11.178
DEGREES OF FREEDOM? 5
RIGHT TAIL VALUE = 4E-05

T VALUE? 0
```

```
10 PRINT "□STUDENT'S T-DISTRIBUTION"
20 PRINT
30 PRINT "(TO END PROGRAM ENTER 0)"
40 PRINT "T VALUE";
50 INPUT T
60 IF T=0 THEN 340
70 PRINT "DEGREES OF FREEDOM";
80 INPUT D
90 X=1
100 Y=1
110 T=T↑2
119 REM - COMPUTE USING INVERSE FOR SMALL T-VALUES
120 IF T<1 THEN 170
130 S=Y
140 R=D
150 Z=T
160 GOTO 200
170 S=D
180 R=Y
190 Z=1/T
200 J=2/9/S
210 K=2/9/R
219 REM - COMPUTE USING APPROXIMATION FORMULAS
220 L=ABS((1-K)*Z↑(1/3)-1+J)/SQR(K*Z↑(2/3)+J)
230 IF R<4 THEN 270
240 X=.5/(1+L*(.196854+L*(.115194+L*(.000344+L*.019527))))↑4
250 X=INT(X*10000+.5)/100000
260 GOTO 290
270 L=L*(1+.08*L↑4/R↑3)
280 GOTO 240
289 REM - ADJUST IF INVERSE WAS COMPUTED
290 IF T>=1 THEN 310
300 X=1-X
310 PRINT "RIGHT TAIL VALUE =";X
320 PRINT
329 REM - RESTART PROGRAM
330 GOTO 40
340 END
```

Student's *t*-distribution Test

This program calculates the *t*-statistic and degrees of freedom for Student's distribution. The calculations can be based on any one of three hypotheses.

The first hypothesis assumes that one population mean is equal to a given value. You must enter the elements of the sample and the value of the mean.

The remaining hypotheses compare two populations. In both tests the means of the two populations are equal, but the standard deviations may be equal or unequal. For these hypotheses you must enter the elements of each sample.

The dimension statement at line 30 limits the size of the samples you may enter. You can change the limit according to the following scheme:

```
30 DIM P(N,2)
```

where *N* = maximum sample size.

Examples:

A sample of children's IQ's was taken, the results being 101, 99, 120, 79, 111, 98, 106, 112, 87, and 97. Calculate the *t*-statistic assuming the population mean is 100.

A second sample was taken, the results being 101, 95, 130, 150, 75, 79, 111, 100, 98 and 91. Calculate the *t*-statistic based on the hypothesis that the two samples have equal means and standard deviations.

```
STUDENT'S T-DISTRIBUTION TEST

TEST 1: MEAN=X
TEST 2: MEAN=MEAN,SD=SD
TEST 3: MEAN=MEAN,SD<>SD

WHICH HYPOTHESIS ? 1

SAMPLE 1 :
 NUMBER OF ELEMENTS ? 10
  ELEMENT 1 ? 101
  ELEMENT 2 ? 99
  ELEMENT 3 ? 120
  ELEMENT 4 ? 79
  ELEMENT 5 ? 111
  ELEMENT 6 ? 98
  ELEMENT 7 ? 106
  ELEMENT 8 ? 112
  ELEMENT 9 ? 87
  ELEMENT 10 ? 97

VALUE OF MEAN? 100

T-VALUE = .26151303
DEGREES OF FREEDOM = 9
```

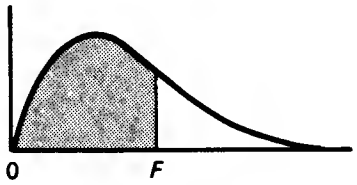
```
STUDENT'S T-DISTRIBUTION TEST

TEST 1: MEAN=X
TEST 2: MEAN=MEAN,SD=SD
TEST 3: MEAN=MEAN,SD<>SD

WHICH HYPOTHESIS ? 2

SAMPLE 1 :
 NUMBER OF ELEMENTS ? 10
  ELEMENT 1 ? 101
  ELEMENT 2 ? 99
  ELEMENT 3 ? 120
  ELEMENT 4 ? 79
  ELEMENT 5 ? 111
  ELEMENT 6 ? 98
  ELEMENT 7 ? 106
  ELEMENT 8 ? 112
  ELEMENT 9 ? 87
  ELEMENT 10 ? 97
SAMPLE 2 :
 NUMBER OF ELEMENTS ? 10
  ELEMENT 1 ? 101
  ELEMENT 2 ? 95
  ELEMENT 3 ? 130
  ELEMENT 4 ? 150
  ELEMENT 5 ? 75
  ELEMENT 6 ? 79
  ELEMENT 7 ? 111
  ELEMENT 8 ? 100
  ELEMENT 9 ? 98
  ELEMENT 10 ? 91


T-VALUE = .24651528
DEGREES OF FREEDOM = 18
```

```
10 PRINT "□STUDENT'S T-DISTRIBUTION TEST"
20 PRINT
29 REM - LIMIT SAMPLE SIZE TO P(N,2) WHERE N=MAXIMUM SAMPLE SIZE
30 DIM P(10,2)
40 DIM V(2),R(2),M(2),D(2)
50 PRINT "TEST 1: MEAN=X"
60 PRINT "TEST 2: MEAN=MEAN,SD=SD"
70 PRINT "TEST 3: MEAN=MEAN,SD<>SD"
80 PRINT:PRINT "WHICH HYPOTHESIS ";
90 INPUT T
100 PRINT
109 REM - INPUT 1 OR 2 SAMPLES DEPENDING ON HYPOTHESIS
110 FOR I=1 TO SGN(T-1)+1
120 V(I)=0
130 D(I)=0
140 PRINT "SAMPLE";I;":"
150 PRINT " NUMBER OF ELEMENTS ";
160 INPUT R(I)
170 FOR J=1 TO R(I)
```

```
180 PRINT "  ELEMENT";J;
190 INPUT P(J,I)
199 REM - ACCUMULATE SAMPLES
200 V(I)=V(I)+P(J,I)
210 D(I)=D(I)+P(J,I)↑2
220 NEXT J
229 REM - COMPUTE INTERMEDIATE VALUES
230 M(I)=V(I)/R(I)
240 V(I)=(D(I)-V(I)↑2/R(I))/(R(I)-1)
250 NEXT I
260 PRINT
270 IF T=2 THEN 340
280 IF T=3 THEN 380
289 REM - INPUT GIVEN VALUE FOR FIRST HYPOTHESIS
290 PRINT "VALUE OF MEAN";
300 INPUT M
309 REM - COMPUTE T AND DEGREES OF FREEDOM FOR FIRST HYPOTHESIS
310 A=(M(1)-M)*SQR(R(1)/V(1))
320 B=R(1)-1
330 GOTO 420
339 REM - COMPUTE T AND DEGREES OF FREEDOM FOR SECOND HYPOTHESIS
340 A=(M(1)-M(2))/SQR(1/R(1)+1/R(2))
350 B=R(1)+R(2)-2
360 A=A/SQR(((R(1)-1)*V(1)+(R(2)-1)*V(2))/B)
370 GOTO 420
379 REM - COMPUTE T AND DEGREES OF FREEDOM FOR THIRD HYPOTHESIS
380 A=(M(1)-M(2))/SQR(V(1)/R(1)+V(2)/R(2))
390 B=(V(1)/R(1)+V(2)/R(2))↑2
400 B=B/((V(1)/R(1))↑2/(R(1)+1)+(V(2)/R(2))↑2/(R(2)+1))-2
410 B=INT(B+.5)
420 PRINT
430 PRINT "T-VALUE =";ABS(A)
440 PRINT "DEGREES OF FREEDOM =";B
450 END
```

F -distribution

This program calculates percentile values for given values on an *F* -distribution curve. You must provide the value of *F,* the degrees of freedom in the numerator and the degrees of freedom in the denominator.



The *F* -distribution

The area of the shaded region represents the percentile.

The *F* -distribution function is approximated using the following formula:

$$\text{percentile} = 1 - \frac{1}{2}(1 + a_1 y + a_2 y^2 + a_3 y^3 + a_4 y^4)^{-4} + \epsilon(y)$$

where: $a_1 = .196854$
$a_2 = .115194$
$a_3 = .000344$
$a_4 = .019527$

$$y = \left(F^{1/3}\left(1 - \frac{2}{9d_2}\right) - \left(1 - \frac{2}{9d_1}\right)\right)\left(\frac{2}{9d_1} + F^{2/3} \cdot \frac{2}{9d_2}\right)^{-1/2}$$

d_1 = degrees of freedom in numerator
d_2 = degrees of freedom in denominator
$|\epsilon(y)| < 2.5 \times 10^{-4}$

Examples:

What is the percentile on an *F* -distribution curve when the *F* -value is .474 and the degrees of freedom are 1 and 18?

What is the percentile when the *F* -value is 23.7 and the degrees of freedom are 3 and 6?

```
F-DISTRIBUTION

(TO END PROGRAM ENTER 0)
F-VALUE? .474
DEGREES OF FREEDOM IN NUMERATOR? 1
DEGREES OF FREEDOM IN DENOMINATOR? 18
PERCENTILE = .4937

F-VALUE? 23.7
DEGREES OF FREEDOM IN NUMERATOR? 3
DEGREES OF FREEDOM IN DENOMINATOR? 6
PERCENTILE = .9984

F-VALUE? 0
```

```
10 PRINT "F-DISTRIBUTION"
20 PRINT
30 PRINT "(TO END PROGRAM ENTER 0)"
40 PRINT "F-VALUE";
50 INPUT F
60 IF F=0 THEN 340
70 PRINT "DEGREES OF FREEDOM ";
71 PRINT "IN NUMERATOR";
80 INPUT D1
90 PRINT "DEGREES OF FREEDOM ";
91 PRINT "IN DENOMINATOR";
100 INPUT D2
110 X=1
119 REM - COMPUTE USING INVERSE FOR SMALL F-VALUES
120 IF F<1 THEN 170
130 S=D1
140 T=D2
150 Z=F
160 GOTO 200
170 S=D2
180 T=D1
190 Z=1/F
200 J=2/9/S
210 K=2/9/T
219 REM - COMPUTE USING APPROXIMATION FORMULAS
220 Y=ABS((1-K)*Z↑(1/3)-1+J)/SQR(K*Z↑(2/3)+J)
230 IF T<4 THEN 270
240 X=.5/(1+Y*(.196854+Y*(.115194+Y*(.000344+Y*.019527))))↑4
250 X=INT(X*10000+.5)/10000
260 GOTO 290
270 Y=Y*(1+.08*Y↑4/T↑3)
280 GOTO 240
289 REM - ADJUST IF INVERSE WAS COMPUTED
290 IF F>=1 THEN 310
300 X=1-X
310 PRINT "PERCENTILE =";1-X
320 PRINT
329 REM - RESTART PROGRAM
330 GOTO 40
340 END
```

OPTION

You may prefer to compute the tail-end value (the area of the unshaded region in the figure above). The program changes necessary are listed following the examples below.

Examples:

What is the tail-end value on an *F* -distribution curve when the *F* -value is .474 and the degrees of freedom are 1 and 18?

What is the tail-end value when the *F* -value is 23.7 and the degrees of freedom are 3 and 6?

```
F-DISTRIBUTION

(TO END PROGRAM ENTER 0)
F-VALUE? .474
DEGREES OF FREEDOM IN NUMERATOR? 1
DEGREES OF FREEDOM IN DENOMINATOR? 18
TAIL END VALUE = .5063

F-VALUE? 23.7
DEGREES OF FREEDOM IN NUMERATOR? 3
DEGREES OF FREEDOM IN DENOMINATOR? 6
TAIL END VALUE = 1.6E-03

F-VALUE? 0
```

```
1 REM - OPTION 310
10 PRINT "F-DISTRIBUTION"
 .
 .
 .
300 X=1-X
310 PRINT "TAIL END VALUE =";X
320 PRINT
329 REM - RESTART PROGRAM
330 GOTO 40
340 END
```

Linear Correlation Coefficient

This program computes the coefficient of correlation between two variables. A linear relationship is assumed between the variables. You must enter the coordinates of a group of data points forming the regression line.

Example:

The height of twelve men and their sons is recorded in the table below. What is the coefficient of correlation between the heights of fathers and the heights of their sons?

father	65	63	67	64	68	62	70	66	68	67	69	71
son	68	66	68	65	69	66	68	65	71	67	68	70

height in inches

```
LINEAR CORRELATION COEFFICIENT

NUMBER OF POINTS? 12
X,Y OF POINT 1 ? 65,68
X,Y OF POINT 2 ? 63,66
X,Y OF POINT 3 ? 67,68
X,Y OF POINT 4 ? 64,65
X,Y OF POINT 5 ? 68,69
X,Y OF POINT 6 ? 62,66
X,Y OF POINT 7 ? 70,68
X,Y OF POINT 8 ? 66,65
X,Y OF POINT 9 ? 68,71
X,Y OF POINT 10 ? 67,67
X,Y OF POINT 11 ? 69,68
X,Y OF POINT 12 ? 71,70

COEFFICIENT OF CORRELATION = .702652258
```

```
10 PRINT "□LINEAR CORRELATION COEFFICIENT"
20 PRINT
30 PRINT "NUMBER OF POINTS";
40 INPUT N
99 REM - ENTER COORDINATES OF DATA POINTS
100 FOR I=1 TO N
110 PRINT "X,Y OF POINT";I;
120 INPUT X,Y
129 REM - ACCUMULATE INTERMEDIATE VALUES
130 J=J+X
140 K=K+Y
150 L=L+X↑2
160 M=M+Y↑2
170 R=R+X*Y
180 NEXT I
189 REM - CALCULATE COEFFICIENT, PRINT
```

```
190 R2=(N*R-J*K)/SQR((N*L-J↑2)*(N*M-K↑2))
200 PRINT
210 PRINT "CORRELATION COEFFICIENT =";
215 PRINT R2
220 END
```

Linear Regression

This program fits a straight line to a given set of coordinates using the method of least squares. The equation of the line, coefficient of determination, coefficient of correlation and standard error of estimate are printed. Once the line has been fitted, you may predict values of y for given values of x.

Example:

The table below shows the height and weight of 11 male college students. Fit a curve to these points. How much would the average 70" and 72" male student weigh?

height (in.)	71	73	64	65	61	70	65	72	63	67	64
weight (lbs.)	160	183	154	168	159	180	145	210	132	168	141

```
LINEAR REGRESSION

NUMBER OF KNOWN POINTS? 11
X,Y OF POINT 1 ? 71,160
X,Y OF POINT 2 ? 73,183
X,Y OF POINT 3 ? 64,154
X,Y OF POINT 4 ? 65,168
X,Y OF POINT 5 ? 61,159
X,Y OF POINT 6 ? 70,180
X,Y OF POINT 7 ? 65,145
X,Y OF POINT 8 ? 72,210
X,Y OF POINT 9 ? 63,132
X,Y OF POINT 10 ? 67,168
X,Y OF POINT 11 ? 64,141

F(X)=-106.791727 + ( 4.04722312 * X)

COEFFICIENT OF DETERMINATION(R12)=  .556260313

COEFFICIENT OF CORRELATION= .745828608

STANDARD ERROR OF ESTIMATE= 15.4134854

INTERPOLATION: (ENTER X=0 TO END)
X =? 70
Y = 176.513892

X =? 72
Y = 184.608338

X =? 0
```

```
10 PRINT "LINEAR REGRESSION"
20 PRINT
30 PRINT "NUMBER OF KNOWN POINTS";
40 INPUT N
99 REM - LOOP TO ENTER COORDINATES OF POINTS
100 FOR I=1 TO N
110 PRINT "X,Y OF POINT";I;
120 INPUT X,Y
129 REM - ACCUMULATE INTERMEDIATE SUMS
130 J=J+X
140 K=K+Y
150 L=L+X↑2
160 M=M+Y↑2
170 R2=R2+X*Y
180 NEXT I
189 REM - COMPUTE CURVE COEFFICIENT
190 B=(N*R2-K*J)/(N*L-J↑2)
200 A=(K-B*J)/N
210 PRINT
220 PRINT "F(X)=";A;"+ (";B;"* X)"
229 REM - COMPUTE REGRESSION ANAYLYSIS
230 J=B*(R2-J*K/N)
240 M=M-K↑2/N
250 K=M-J
260 PRINT
270 R2=J/M
280 PRINT "COEFFICIENT OF ";
281 PRINT "DETERMINATION";
282 PRINT "(R↑2)= ";R2
283 PRINT
290 PRINT "COEFFICIENT OF ";
291 PRINT "CORRELATION=";SQR(R2)
292 PRINT
300 PRINT"STANDARD ERROR OF ESTIMATE=";
301 PRINT SQR(K/(N-2))
310 PRINT
319 REM - ESTIMATE Y-COORDINATES OF POINTS WITH ENTERED X-COORDINATES
320 PRINT "INTERPOLATION: ";
321 PRINT "(ENTER X=0 TO END)"
330 PRINT "X =";
340 INPUT X
349 REM - RESTART OR END PROGRAM? USER INPUT REQUIRED
350 IF X=0 THEN 390
360 PRINT "Y =";A+B*X
370 PRINT
380 GOTO 330
390 END
```

Multiple Linear Regression

This program finds the coefficients of a multiple variable linear equation using the method of least squares. The equation is of the following form:

$$y = c + a_1 x_1 + a_2 x_2 + \ldots a_n x_n$$

where: y = dependent variable
c = constant
$a_1, a_2 \ldots a_n$ = coefficients of independent variables $x_1, x_2, \ldots x_n$

The constant and the coefficients are printed.

You must provide the x - and y -coordinates of known data points. Once the equation has been found using the data you enter, you may predict values of the dependent variables for given values of the independent variables.

The dimension statement at line 30 limits the number of known data points the equation may contain. You can change this limit according to the following scheme:

```
30 DIM X(N+1), S(N+1), T(N+1), A(N+1,N+2)
```

where N = the number of known data points.

Example:

The table below shows the age, height and weight of eight boys. Using weight as the dependent variable, fit a curve to the data. Estimate the weight of a seven year old boy who is 51 inches tall.

age	8	9	6	10	8	9	9	7
height	48	49	44	59	55	51	55	50
weight	59	55	50	80	61	75	67	58

```
MULTIPLE LINEAR REGRESSION

NUMBER OF KNOWN POINTS? 8
# OF INDEPENDENT VARIABLES? 2
POINT 1
 VARIABLE 1 ? 8
 VARIABLE 2 ? 48
 DEPENDENT VARIABLE? 59
POINT 2
 VARIABLE 1 ? 9
 VARIABLE 2 ? 49
 DEPENDENT VARIABLE? 55
```

```
POINT 3
 VARIABLE 1 ? 6
 VARIABLE 2 ? 44
 DEPENDENT VARIABLE? 50
POINT 4
 VARIABLE 1 ? 10
 VARIABLE 2 ? 59
 DEPENDENT VARIABLE? 80
POINT 5
 VARIABLE 1 ? 8
 VARIABLE 2 ? 55
 DEPENDENT VARIABLE? 61
POINT 6
 VARIABLE 1 ? 9
 VARIABLE 2 ? 51
 DEPENDENT VARIABLE? 75
POINT 7
 VARIABLE 1 ? 9
 VARIABLE 2 ? 55
 DEPENDENT VARIABLE? 67
POINT 8
 VARIABLE 1 ? 7
 VARIABLE 2 ? 50
 DEPENDENT VARIABLE? 58

EQUATION COEFFICIENTS:
     CONSTANT:-15.7021277
VARIABLE( 1 ): 3.68085106
VARIABLE( 2 ): .943262412

COEFFICIENT OF DETERMINATION
          (R↑2) = .715697404
COEFFICIENT OF MULTIPLE
CORRELATION = .845989009
STANDARD ERROR OF ESTIMATE = 6.42887917

INTERPOLATION: (ENTER 0 TO END PROGRAM)
VARIABLE 1 ? 7
VARIABLE 2 ? 51
DEPENDENT VARIABLE = 58.1702128

VARIABLE 1 ? 0
```

```
10 PRINT "□MULTIPLE LINEAR REGRESSION"
20 PRINT
29 REM - SET ARRAY LIMITS TO X(N+1), S(N+1), T(N+1), A(N+1,N+2)
30 DIM X(9),S(9),T(9),A(9,10)
40 PRINT "NUMBER OF KNOWN POINTS";
50 INPUT N
60 PRINT "# OF INDEPENDENT VARIABLES";
70 INPUT V
80 X(1)=1
90 FOR I=1 TO N
100 PRINT "POINT";I
110 FOR J=1 TO V
119 REM - ENTER INDEPENDENT VARIABLES FOR EACH POINT
```

```
120 PRINT " VARIABLE";J;
130 INPUT X(J+1)
140 NEXT J
149 REM - ENTER DEPENDENT VARIABLE FOR EACH POINT
150 PRINT " DEPENDENT VARIABLE";
160 INPUT X(V+2)
169 REM - POPULATE A MATRIX TO BE USED IN CURVE FITTING
170 FOR K=1 TO V+1
180 FOR L=1 TO V+2
190 A(K,L)=A(K,L)+X(K)*X(L)
200 S(K)=A(K,V+2)
210 NEXT L
220 NEXT K
230 S(V+2)=S(V+2)+X(V+2)↑2
240 NEXT I
248 REM - STATEMENTS 250 TO 500 FIT CURVE BY SOLVING THE SYSTEM OF;
249 REM - LINEAR EQUATIONS IN MATRIX A()
250 FOR I=2 TO V+1
260 T(I)=A(1,I)
270 NEXT I
280 FOR I=1 TO V+1
290 J=I
300 IF A(J,I)<>0 THEN 340
305 J=J+1
310 IF J<=V+1 THEN 300
320 PRINT "NO UNIQUE SOLUTION"
330 GOTO 810
340 FOR K=1 TO V+2
350 B=A(I,K)
360 A(I,K)=A(J,K)
370 A(J,K)=B
380 NEXT K
390 Z=1/A(I,I)
400 FOR K=1 TO V+2
410 A(I,K)=Z*A(I,K)
420 NEXT K
430 FOR J=1 TO V+1
440 IF J=I THEN 490
450 Z=-A(J,I)
460 FOR K=1 TO V+2
470 A(J,K)=A(J,K)+Z*A(I,K)
480 NEXT K
490 NEXT J
500 NEXT I
510 PRINT
520 PRINT "EQUATION COEFFICIENTS:"
525 PRINT "      CONSTANT:"A(1,V+2)
530 FOR I=2 TO V+1
540 PRINT "VARIABLE(";I-1;"):";A(I,V+2)
550 NEXT I
560 P=0
570 FOR I=2 TO V+1
580 P=P+A(I,V+2)*(S(I)-T(I)*S(1)/N)
590 NEXT I
600 R=S(V+2)-S(1)↑2/N
610 Z=R-P
620 L=N-V-1
```

```
630 I=P/V
640 PRINT
650 I=P/R
660 PRINT "COEFFICIENT OF ";
661 PRINT "DETERMINATION "
665 PRINT "            (R12) =";I
670 PRINT "COEFFICIENT OF MULTIPLE"
675 PRINT "CORRELATION =";SQR(I)
680 PRINT "STANDARD ERROR OF ESTIMATE";
681 PRINT SQR(ABS(Z/L))
690 PRINT
699 REM - ESTIMATE DEPENDENT VARIABLE FROM ENTERED INDEPENDENT VARABLES
700 PRINT "INTERPOLATION: ";
701 PRINT "(ENTER 0 TO END PROGRAM)"
710 P=A(1,V+2)
720 FOR J=1 TO V
730 PRINT "VARIABLE";J;
740 INPUT X
749 REM - TEST FOR END OF PROGRAM
750 IF X=0 THEN 810
760 P=P+A(J+1,V+2)*X
770 NEXT J
780 PRINT "DEPENDENT VARIABLE =";P
790 PRINT
799 REM - RETURN FOR MORE DATA
800 GOTO 710
810 END
```

N th Order Regression

This program finds the coefficients of an *N* th order equation using the method of least squares. The equation is of the following form:

$$y = c + a_1 x + a_2 x^2 + \ldots a_n x^n$$

where: y = dependent variable
c = constant
$a_1, a_2 \ldots a_n$ = coefficients of independent variables $x, x^2, \ldots x^n$, respectively

The equation coefficients, coefficient of determination, coefficient of correlation and standard error of estimate are printed.

You must provide the *x* - and *y* -coordinates for known data points. Once the equation has been computed you may predict values of *y* for given values of *x*.

The dimension statement at line 30 limits the degree of the equation. You can change this limit according to the following scheme:

```
30 DIM A(2*D+1), R(D+1,D+2), T(D+2)
```

where D = maximum degree of equation.

Example:

The table below gives the stopping distance (reaction plus braking distance) of an automobile at various speeds. Fit an exponential curve to the data. Estimate the stopping distance at 55 m.p.h.

m.p.h.	20	30	40	50	60	70
stopping distance	54	90	138	206	292	396

```
30 DIM A(5),R(3,4),T(4)
RUN


NTH-ORDER REGRESSION

DEGREE OF EQUATION? 2
NUMBER OF KNOWN POINTS? 6
X,Y OF POINT 1 ? 20,54
X,Y OF POINT 2 ? 30,90
X,Y OF POINT 3 ? 40,138
X,Y OF POINT 4 ? 50,206
X,Y OF POINT 5 ? 60,292
X,Y OF POINT 6 ? 70,396

              CONSTANT = 41.7714472
 1 DEGREE COEFFICIENT =-1.09571524
 2 DEGREE COEFFICIENT = .0878571531
```

```
COEFFICIENT OF DETERMINATION
(R↑2) = .999927959
CORRELATION COEFFICIENT = .999963979
STANDARD ERROR ESTIMATE = 1.42094106

INTERPOLATION: (ENTER 0 TO END)
X =? 55
Y = 247.274998

X =? 0
```

```
10 PRINT "□N'TH ORDER REGRESSION"
20 PRINT
28 REM - SET LIMITS OF DEGREE OF EQUATION TO A(2D+1),R(D+1,D+2),T(D+2)
29 REM - (WHERE D=MAXIMUM DEGREE OF EQUATION)
30 DIM A(13),R(7,8),T(8)
40 PRINT "DEGREE OF EQUATION";
50 INPUT D
60 PRINT "NUMBER OF KNOWN POINTS";
70 INPUT N
80 A(1)=N
89 REM - ENTER COORDINATES OF DATA POINTS
90 FOR I=1 TO N
100 PRINT "X,Y OF POINT";I;
110 INPUT X,Y
119 REM - LINES 120 TO 200 POPULATE MATRICES WITH A SYSTEM OF EQUATIONS
120 FOR J=2 TO 2*D+1
130 A(J)=A(J)+X↑(J-1)
140 NEXT J
150 FOR K=1 TO D+1
160 R(K,D+2)=T(K)+Y*X↑(K-1)
170 T(K)=T(K)+Y*X↑(K-1)
180 NEXT K
190 T(D+2)=T(D+2)+Y↑2
200 NEXT I
209 REM - LINES 210 TO 490 SOLVE THE SYSTEM OF EQUATIONS IN THE MATRICES
210 FOR J=1 TO D+1
220 FOR K=1 TO D+1
230 R(J,K)=A(J+K-1)
240 NEXT K
250 NEXT J
260 FOR J=1 TO D+1
270 FOR K=J TO D+1
280 IF R(K,J)<>0 THEN 320
290 NEXT K
300 PRINT "NO UNIQUE SOLUTION"
310 GOTO 790
320 FOR I=1 TO D+2
330 S=R(J,I)
340 R(J,I)=R(K,I)
350 R(K,I)=S
360 NEXT I
370 Z=1/R(J,J)
380 FOR I=1 TO D+2
390 R(J,I)=Z*R(J,I)
400 NEXT I
```

```
410 FOR K=1 TO D+1
420 IF K=J THEN 470
430 Z=-R(K,J)
440 FOR I=1 TO D+2
450 R(K,I)=R(K,I)+Z*R(J,I)
460 NEXT I
470 NEXT K
480 NEXT J
490 PRINT
495 PRINT "              CONSTANT =";
496 PRINT R(1,D+2)
499 REM - PRINT EQUATION COEFFICIENTS
500 FOR J=1 TO D
510 PRINT J;"DEGREE COEFFICIENT =";
511 PRINT R(J+1,D+2)
520 NEXT J
530 PRINT
539 REM - COMPUTE REGRESSION ANALYSIS
540 P=0
550 FOR J=2 TO D+1
560 P=P+R(J,D+2)*(T(J)-A(J)*T(1)/N)
570 NEXT J
580 Q=T(D+2)-T(1)↑2/N
590 Z=Q-P
600 I=N-D-1
620 PRINT
630 J=P/Q
640 PRINT "COEFFICIENT OF ";
641 PRINT "DETERMINATION"
645 PRINT "(R↑2) =";J
650 PRINT "CORRELATION COEFFICIENT =";
651 PRINT SQR(J)
660 PRINT "STANDARD ERROR ESTIMATE =";
661 PRINT SQR(Z/I)
670 PRINT
679 REM - COMPUTE Y-COORDINATE FROM ENTERED X-COORDINATE
680 PRINT "INTERPOLATION: ";
681 PRINT "(ENTER 0 TO END)"
690 P=R(1,D+2)
700 PRINT "X =";
710 INPUT X
720 IF X=0 THEN 790
730 FOR J=1 TO D
740 P=P+R(J+1,D+2)*X↑J
750 NEXT J
760 PRINT "Y =";P
770 PRINT
780 GOTO 690
790 END
```

Geometric Regression

This program fits a geometric curve to a set of coordinates using the method of least squares. The equation, coefficient of determination, coefficient of correlation and standard error of estimate are printed.

You must provide the x - and y -coordinates of known data points. Once the curve has been fitted you may predict values of y for given values of x.

Example:

The table below gives the pressures of a gas measured at various volumes in an experiment. The relationship between pressure and volume of a gas is expressed by the following formula:

$$PV^K = C$$

where. P = pressure
V = volume
C and K are constants.

This formula can be rewritten in standard geometric form:

$$P = CV^{-K}$$

Note the exponent is negative, which accounts for the negative exponents the program calculates.

Fit a geometric curve to the data and estimate the pressure of 90 cubic inches of the gas.

volume	56.1	60.7	73.2	88.3	120.1	187.5
pressure	57.0	51.0	39.2	30.2	19.6	10.5

```
GEOMETRIC REGRESSION

NUMBER OF KNOWN POINTS? 6
X,Y OF POINT 1 ? 56.1,57.0
X,Y OF POINT 2 ? 60.7,51.0
X,Y OF POINT 3 ? 73.2,39.2
X,Y OF POINT 4 ? 88.3,30.2
X,Y OF POINT 5 ? 120.1,19.6
X,Y OF POINT 6 ? 187.5,10.5

F(X) = 16103.7139 * X↑-1.40155091

COEFFICIENT OF DETERMINATION
(R↑2) = .999999206

COEFFICIENT OF CORRELATION = .999999603

STANDARD ERROR OF ESTIMATE =  6.37591016E-04
```

```
INTERPOLATION: (X=0 TO END)
X =? 90
Y = 29.3734983

X =? 0
```

```
10 PRINT "□GEOMETRIC REGRESSION"
20 PRINT
30 PRINT "NUMBER OF KNOWN POINTS";
40 INPUT N
99 REM - ENTER COORDINATES OF DATA POINTS
100 FOR I=1 TO N
110 PRINT "X,Y OF POINT";I;
120 INPUT X,Y
129 REM - ACCUMULATE INTERMEDIATE VALUES
130 Y=LOG(Y)
140 X=LOG(X)
150 J=J+X
160 K=K+Y
170 L=L+X↑2
180 M=M+Y↑2
190 R2=R2+X*Y
200 NEXT I
209 REM - CALCULATE AND PRINT COEFFICIENTS OF EQUATION
210 B=(N*R2-K*J)/(N*L-J↑2)
220 A=(K-B*J)/N
230 PRINT
240 PRINT "F(X) =";EXP(A);"* X↑";B
249 REM - CALCULATE REGRESSION ANALYSIS
250 J=B*(R2-J*K/N)
260 M=M-K↑2/N
270 K=M-J
280 PRINT
290 R2=J/M
300 PRINT "COEFFICIENT OF ";
301 PRINT "DETERMINATION"
305 PRINT "(R↑2) =";R2
306 PRINT
310 PRINT "COEFFICIENT OF ";
311 PRINT "CORRELATION =";SQR(R2)
315 PRINT
320 PRINT "STANDARD ERROR OF ";
325 PRINT "ESTIMATE = ";SQR(K/(N-2))
326 PRINT
330 PRINT
339 REM - ESTIMATE Y-COORDINATE FROM ENTERED X-COORDINATE
340 PRINT "INTERPOLATION: ";
341 PRINT "(X=0 TO END)"
350 PRINT "X =";
360 INPUT X
369 REM - RESTART OR END PROGRAM? USER INPUT REQUIRED
370 IF X=0 THEN 410
380 PRINT "Y =";EXP(A)*X↑B
390 PRINT
399 REM - RETURN FOR MORE DATA
400 GOTO 350
410 END
```

Exponential Regression

This program finds the coefficients of an equation for an exponential curve. The equation is in the following form:

$$f(x) = ae^{bx}$$

where a and b are the calculated coefficients.

The equation coefficients, coefficient of determination, coefficient of correlation and standard error of estimate are printed.

You must provide the x - and y -coordinates for known data points. Once the curve has been fitted you may predict values of y for given values of x.

Example:

The table below shows the number of bacteria present in a culture at various points in time. Fit an exponential curve to the data and estimate the number of bacteria after 7 hours.

number of hours	0	1	2	3	4	5	6
number of bacteria	25	38	58	89	135	206	315

```
EXPONENTIAL REGRESSION

NUMBER OF KNOWN POINTS? 7
X,Y OF POINT 1 ? 0,25
X,Y OF POINT 2 ? 1,38
X,Y OF POINT 3 ? 2,58
X,Y OF POINT 4 ? 3,89
X,Y OF POINT 5 ? 4,135
X,Y OF POINT 6 ? 5,206
X,Y OF POINT 7 ? 6,315

A = 24.9616634
B = .422375081

COEFFICIENT OF DETERMINATION
(R12) = .999993572
COEFFICIENT OF CORRELATION = .999996779
STANDARD ERROR OF ESTIMATE = 2.53424554E-03

INTERPOLATION: (X=0 TO END)
X =? 7
Y= 480.086716

X =? 0
```

```
10 PRINT "EXPONENTIAL REGRESSION"
20 PRINT
30 PRINT "NUMBER OF KNOWN POINTS";
40 INPUT N
50 J=0
60 K=0
70 L=0
80 M=0
90 R2=0
99 REM - ENTER COORDINATES OF DATA POINTS
100 FOR I=1 TO N
110 PRINT "X,Y OF POINT";I;
120 INPUT X,Y
129 REM - ACCUMULATE INTERMEDIATE VALUES
130 Y=LOG(Y)
140 J=J+X
150 K=K+Y
160 L=L+X↑2
170 M=M+Y↑2
180 R2=R2+X*Y
190 NEXT I
199 REM - CALCULATE AND PRINT COEFFICIENTSOF EQUATION
200 B=(N*R2-K*J)/(N*L-J↑2)
210 A=(K-B*J)/N
220 PRINT
230 PRINT"A =";EXP(A)
240 PRINT "B =";B
249 REM - CALCULATE REGRESSION TABLE VALUES
250 J=B*(R2-J*K/N)
260 M=M-K↑2/N
270 K=M-J
280 PRINT
290 R2=J/M
300 PRINT "COEFFICIENT OF ";
301 PRINT "DETERMINATION"
305 PRINT "(R↑2) =";R2
306 PRINT
310 PRINT "COEFFICIENT OF ";
311 PRINT "CORRELATION =";
315 PRINT SQR(R2)
316 PRINT
320 PRINT "STANDARD ERROR OF ";
321 PRINT "ESTIMATE =";SQR(K/(N-2))
326 PRINT
330 PRINT
339 REM - ESTIMATE Y-VALUE FROM ENTERED X-VALUE
340 PRINT "INTERPOLATION: ";
341 PRINT "(X=0 TO END)"
350 PRINT "X =";
360 INPUT X
370 IF X=0 THEN 410
380 PRINT "Y =";EXP(A)*EXP(B*X)
390 PRINT
399 REM - RETURN FOR MORE DATA
400 GOTO 350
410 END
```

System Reliability

This program calculates the reliability of an operating system that is subject to wearout and chance failure. You must enter the system's operating time and the wearout time and failure rate of each component.

Example:

Compute the reliability of a computer system operating for 1000 hours with the components shown in the list below.

	wearout (hrs.)	failure
CPU	15,000	.00020
terminal	3,000	.00010
disk	3,000	.00015
printer	1,500	.00015

```
SYSTEM RELIABILITY

(TO END PROGRAM ENTER 0)
OPERATING TIME IN HOURS? 1000
NUMBER OF COMPONENTS? 4
COMPONENT 1
  AVERAGE WEAROUT TIME? 15000
  AVERAGE FAILURE RATE? .0002
COMPONENT 2
  AVERAGE WEAROUT TIME? 3000
  AVERAGE FAILURE RATE? .0001
COMPONENT 3
  AVERAGE WEAROUT TIME? 3000
  AVERAGE FAILURE RATE? .00015
COMPONENT 4
  AVERAGE WEAROUT TIME? 1500
  AVERAGE FAILURE RATE? .00015

SYSTEM RELIABILITY = .135335233

OPERATING TIME IN HOURS? 0
```

```
10 PRINT "□SYSTEM RELIABILITY"
20 PRINT
30 PRINT "(TO END PROGRAM ENTER 0)"
40 PRINT "OPERATING TIME IN HOURS";
50 INPUT T
59 REM - TEST FOR END OF PROGRAM
60 IF T=0 THEN 230
70 PRINT "NUMBER OF COMPONENTS";
```

```
80 INPUT N
90 Z=0
99 REM - ENTER DATA FOR EACH COMPONENT
100 FOR I=1 TO N
105 PRINT
110 PRINT "COMPONENT";I
120 PRINT "AVERAGE WEAROUT TIME";
130 INPUT W
140 PRINT "AVERAGE FAILURE RATE";
150 INPUT F
159 REM - INCLUDE EACH COMPONENT IN RELIABILITY
160 Z=Z+1/W+F
170 NEXT I
180 PRINT
189 REM - CALCULATE RELIABILITY, PRINT
190 Z=EXP(-Z*T)
200 PRINT "SYSTEM RELIABILITY =";Z
210 PRINT
219 REM - RESTART PROGRAM
220 GOTO 40
230 END
```

Average Growth Rate, Future Projections

This program calculates the average growth rate of a company, then projects figures for future years. The growth rate and projections could be computed for any aspect of a company, such as sales, earnings, number of employees, or patronage. You must provide established figures for a past series of years.

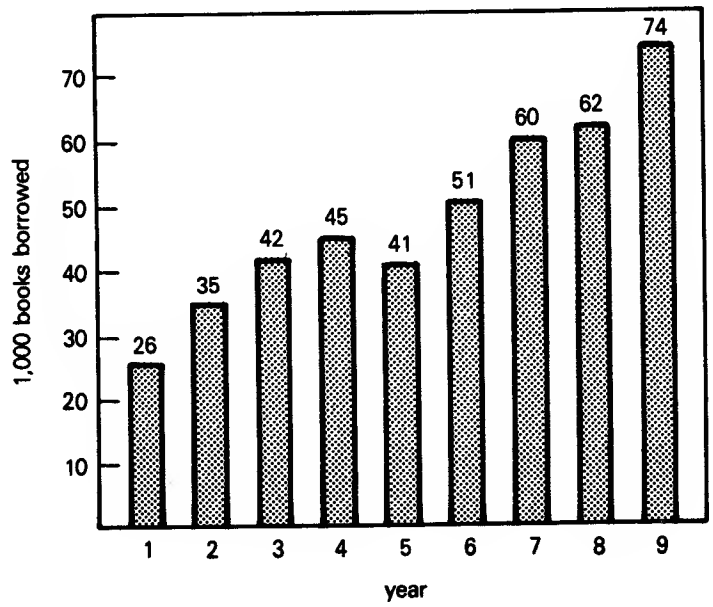
The dimension statement at line 30 limits the number of past figures you may enter. Any need to alter this limit should be done in the following manner:

```
30 DIM S(N)
```

where *N* = the number of years for which figures are known.

Example:

The borrowing records for Claremount County Library are tabulated in the graph below. What is its average growth rate? How many books can it expect to lend in its tenth and twentieth years of service?




```
30 DIM S(9)
RUN
AVERAGE GROWTH RATE, FUTURE PROJECTIONS

NUMBER OF YEARS FIGURES ESTABLISHED? 9
FIGURE: YEAR 1 ? 26
        YEAR 2 ? 35
        YEAR 3 ? 42
        YEAR 4 ? 45
        YEAR 5 ? 41
        YEAR 6 ? 51
        YEAR 7 ? 60
        YEAR 8 ? 62
        YEAR 9 ? 74
AVERAGE GROWTH RATE = 11.88 %
```

```
(ENTER 0 TO END PROGRAM)
PROJECTED SALES FOR YEAR? 10
                                = 81.29
PROJECTED SALES FOR YEAR? 20
                                = 249.88
PROJECTED SALES FOR YEAR? 0
```

```
10 PRINT "□AVERAGE GROWTH RATE,";
15 PRINT " FUTURE PROJECTIONS"
20 PRINT
29 REM - SET ARRAY S TO THE NUMBER OF YEARS FOR WHICH FIGURES ARE KNOWN
30 DIM S(20)
40 PRINT "NUMBER OF YEARS FIGURES";
45 PRINT " ESTABLISHED";
50 INPUT N
60 FOR I=1 TO N
70 IF I>1 THEN 100
80 PRINT "FIGURE: YEAR";I;
90 GOTO 110
100 PRINT "          YEAR";I;
110 INPUT S(I)
120 NEXT I
129 REM - INITIALIZE VARIABLES FOR FIRST YEAR
130 T=LOG(S(1))
140 V=0
149 REM - LOOP FOR REMAINING YEARS OF HISTORY
150 FOR I=2 TO N
160 L=LOG(S(I))
170 T=T+L
180 V=V+(I-1)*L
190 NEXT I
199 REM - CALCULATE AVERAGE GROWTH RATE
200 A=6*(2*V/(N-1)-T)/(N)/(N+1)
210 G=EXP(A)-1
219 REM - ROUND OFF, PRINT
220 PRINT "AVERAGE GROWTH RATE =";
225 PRINT INT(G*10000+.5)/100;"%"
230 PRINT
239 REM - CALCULATE AVERAGE ANNUAL GROWTH FACTOR
240 S=EXP(T/N-A*(N-1)/2)
250 PRINT "(ENTER 0 TO END PROGRAM)"
259 REM - INPUT YEAR NUMBER
260 PRINT "PROJECTED SALES FOR YEAR";
270 INPUT Y1
279 REM - TEST FOR END OF PROGRAM
280 IF Y1=0 THEN 320
289 REM - CALCULATE  PROJECTED SALES FIGURE
290 S1=S*(1+G)↑(Y1-1)
299 REM - ROUND OFF, PRINT
300 PRINT TAB(28);"=";
305 PRINT INT(S1*100+.5)/100
309 REM - RETURN FOR MORE DATA
310 GOTO 260
320 END
```

Federal Withholding Taxes

This program calculates the amount of federal income and FICA taxes withheld from one's earnings. You must provide employee information as to marital status, the number of exemptions claimed, the amount of taxable pay, and year-to-date taxable pay.

The number of pay periods per year is established at line 80. If your pay period is other than monthly, you must alter this statement to set *N* equal to the number of pay periods per year.

There is a considerable amount of tax information which may change from year to year. The values listed in the data tables at lines 30 and 40 are among those that may need periodic revision. The annual values for single and married persons should be compared each year with those listed in Table 7 of the current IRS Circular E.

The annual FICA rate, the FICA cutoff amount and the annual amount of withholding allowance may also need revision. The values established at lines 50, 60 and 70 should also be compared to those listed in the current IRS circular.

Annual rates and cutoffs are used irrespective of your actual pay period frequency. The program automatically adjusts them to match your pay period.

Examples:

Judy earns $900.00 per month. The payroll clerk is figuring her March paycheck. Judy is single and claims only herself as a dependent. What amounts are withheld from her paycheck?

Dr. Berger has earned $1,408.75 this month. So far this year he has grossed $20,188.72. He is married and claims four dependents. What amounts will be withheld this month for the federal government?

```
FEDERAL WITHHOLDING TAXES


MARITAL STATUS (1=SINGLE, 2=MARRIED)? 1
WITHHOLDING TAX EXEMPTIONS? 1
TAXABLE PAY? 900
YTD TAXABLE PAY? 1800
TAXABLE = $ 900
INCOME TAX = $ 128.5
FICA = $ 55.17

MORE DATA (1=YES, 0=NO)? 1

MARITAL STATUS (1=SINGLE, 2=MARRIED)? 2
WITHHOLDING TAX EXEMPTIONS? 4
TAXABLE PAY? 1408.75
YTD TAXABLE PAY? 23750.03
TAXABLE = $ 1408.75
INCOME TAX = $ 152.09
FICA = $ 0

MORE DATA (1=YES, 0=NO)? 0
```

```
10 PRINT "FEDERAL WITHHOLDING TAXES"
20 PRINT
27 REM - THE FOLLOWING DATA CONTAINS THE 1979 TAX TABLES FROM IRS
28 REM - CIRCULAR E, PERCENTAGE METHOD, TABLE 7 (ANNUAL PAYROLL)
29 REM - FOR SINGLE PERSONS
30 DATA 15,1420,18,3300,21,6800,26,10200,30,14200,34,17200,39,22500
39 REM - FOR MARRIED PERSONS
40 DATA 15,2400,18,6600,21,10900,24,15000,28,19200,32,23600,37,28900
49 REM - F1=FICA RATE AS DECIMAL
50 F1=.0613
59 REM - F2=FICA CUTOFF AMOUNT
60 F2=22900
69 REM - W1=AMOUNT OF WITHHOLDING ALLOWANCE (ANNUAL PAYROLL)
70 W1=1000
79 REM - N=NUMBER OF PAY PERIODS PER YEAR
80 N=12
89 REM - LOAD THE TAX TABLE ARRAYS FROM DATA TABLES
90 DIM F1(28)
100 FOR I=1 TO 28
110 READ F1(I)
120 NEXT I
130 PRINT
139 REM - STATEMENTS 140 TO 210 REQUEST PERTINENT EMPLOYEE DATA
140 PRINT "MARITAL STATUS (1=SINGLE,";
141 PRINT " 2=MARRIED)";
150 INPUT S
160 PRINT "WITHHOLDING TAX EXEMPTIONS";
170 INPUT W
180 PRINT "TAXABLE PAY";
190 INPUT P
199 REM - Y=TOTAL TAXABLE PAY THIS YEAR, EXCLUDING CURRENT PAYCHECK
200 PRINT "YTD TAXABLE PAY";
210 INPUT Y
219 REM - ANNUALIZE CURRENT TAXABLE PAY; ADJUST FOR EXEMPTIONS
220 G=P*N-W1*W
230 T1=0
239 REM - CALCULATE INCOME TAX
240 FOR I=2 TO 7
250 X=2*I+14*(S-1)-1
260 IF G<=F1(X-1) THEN 330
270 IF G>F1(X+1)THEN 300
280 T1=T1+(G-F1(X-1))*F1(X-2)/100
290 GOTO 330
300 T1=T1+(F1(X+1)-F1(X-1))*F1(X-2)/100
310 NEXT I
320 T1=T1+(G-F1(X+1))*F1(X)/100
329 REM - ROUND OFF TO NEAREST CENT
330 T1=INT((T1/N)*100+.5)/100
340 T2=0
349 REM - CALCULATE FICA
350 IF Y>F2 THEN 400
360 IF Y+P>F2 THEN 390
370 T2=INT((P*F1)*100+.5)/100
380 GOTO 400
389 REM - ROUND OFF TO NEAREST CENT
390 T2=INT(((F2-Y)*F1)*100+.5)/100
400 PRINT "TAXABLE = $";P
```

```
410 PRINT "INCOME TAX = $";T1
420 PRINT "FICA = $";T2
430 PRINT
439 REM - RESTART OR END PROGRAM?
440 PRINT "MORE DATA (1=YES, 0=NO)";
450 INPUT S
460 IF S=1 THEN 130
470 END
```

Tax Depreciation Schedule

This program tabulates annual depreciation amounts. You can use the sum of digits method or any declining balance percentage method. You must know the purchase price (initial value), salvage value at the end of the depreciable life, and the life of the item being depreciated. If you are doing declining balance depreciation, you must also know the percentage method.

Examples:

The Miracle Corporation put a new roof on their office building for $27,000.00. They expect to replace it in nine years. What would the annual depreciation amounts be using the sum of digits?

Heavenly Bank built a new home office building for $1.2 million. Run a tax depreciation schedule on the building using 150% declining balance method with a 30 year life. Assume a salvage value of $250,000. You will notice that the depreciation falls below straight line ($31,666.67) per year) at year nine.

```
TAX DEPRECIATION SCHEDULE

PURCHASE PRICE? 27000
SALVAGE VALUE? 0
LIFE IN YEARS? 9
ENTER 1 FOR SUM OF DIGITS,
2 FOR DECLINING BALANCE? 1


     SUM OF DIGITS TAX DEPRECIATION
               PRICE $ 27000
        SALVAGE VALUE $ 0
      NET DEPRECIATED $ 27000
              LIFE 9 YEARS

YEAR    DEPRECIATION   BALANCE
 1      5400           21600
 2      4800           16800
 3      4200           12600
 4      3600           9000
 5      3000           6000
 6      2400           3600
 7      1800           1800
 8      1200           600
 9      600            0

MORE DATA? (1=YES, 0=NO)? 1

PURCHASE PRICE? 1200000
SALVAGE VALUE? 250000
LIFE IN YEARS? 30
ENTER 1 FOR SUM OF DIGITS,
2 FOR DECLINING BALANCE? 2
METHOD IN %? 150
```

```
        DECLINING BALANCE TAX
            DEPRECIATION
              PRICE $ 1200000
        SALVAGE VALUE $ 250000
      NET DEPRECIATED $ 950000
              LIFE 30 YEARS
              METHOD  150 %

YEAR    DEPRECIATION   BALANCE

 1      47500          902500
 2      45125          857375
 3      42868.75       814506.25
 4      40725.31       773780.94
 5      38689.05       735091.89
 6      36754.59       698337.3
 7      34916.86       663420.44
 8      33171.02       630249.42
 9      31512.47       598736.95
 10      29936.85      568800.1
 11      28440         540360.1
 12      27018         513342.1
 13      25667.1       487675
 14      24383.75      463291.25
 15      23164.56      440126.69
 16      22006.33      418120.36
 17      20906.02      397214.34
 18      19860.72      377353.62
 19      18867.68      358485.94
 20      17924.3       340561.64
 21      17028.08      323533.56
 22      16176.68      307356.88
 23      15367.84      291989.04
 24      14599.45      277389.59
 25      13869.48      263520.11
 26      13176.01      250344.1
 27      12517.2       237826.9
 28      11891.34      225935.56
 29      11296.78      214638.78
 30      10731.94      203906.84

MORE DATA? (1=YES,0=NO)? 0
```

```
10 PRINT "□TAX DEPRECIATION SCHEDULE"
20 PRINT
29 REM - ENTER INITIAL VALUE AND ROUND OFF TO NEAREST CENT
30 PRINT "PURCHASE PRICE";
40 INPUT V
50 V=INT(V*100+.5)/100
59 REM - ENTER END VALUE AND ROUND OFF TO NEAREST CENT
60 PRINT "SALVAGE VALUE";
70 INPUT S
80 S=INT(S*100+.5)/100
89 REM - COMPUTE AMOUNT TO DEPRECIATE
90 D=V-S
99 REM - ENTER LENGTH OF DEPRECIATION
```

```
100 PRINT "LIFE IN YEARS";
110 INPUT Y
119 REM - CHOOSE DEPRECIATION METHOD
120 PRINT "ENTER 1 FOR SUM OF DIGITS,"
125 PRINT "2 FOR DECLINING BALANCE";
130 INPUT X
140 IF X=2 THEN 450
150 IF X<>1 THEN 120
158 REM - BY SUM OF DIGITS METHOD
159 REM - R1 IS THE CUMULATIVE AMOUNT DEPRECIATED
160 R1=0
170 N=1
260 PRINT TAB(4);"□   SUM OF DIGITS TAX ";
265 PRINT "DEPRECIATION"
270 PRINT TAB(14);"PRICE $";V
280 PRINT TAB(7);"SALVAGE VALUE $";S
290 PRINT TAB(5);"NET DEPRECIATED $";
291 PRINT V-S
300 PRINT TAB(13);"LIFE";Y;"YEARS"
310 PRINT
320 PRINT "YEAR   DEPRECIATION";
321 PRINT "  BALANCE"
322 N=11
325 FOR I=1 TO Y
326 REM - CHECK FOR FULL SCREEN (23 LINES)
327 IF N/23=INT(N/23) THEN GOSUB 800
329 REM - COMPUTE DEPRECIATION AND ROUND OFF TO NEAREST CENT
330 R=2*D*(Y-I+1)/((Y+1)*Y)
340 R=INT(R*100+.5)/100
349 REM - ACCUMULATE DEPRECIATION
350 R1=R1+R
359 REM - COMPUTE BALANCE TO DEPRECIATE
360 B=D-R1
369 REM - TEST FOR COMPLETE DEPRECIATION
370 IF B>=0 THEN 410
380 R1=R1+R
400 B=0
410 PRINT I;"    ";R,B
420 N=N+1
430 NEXT I
440 GOTO 700
448 REM - BY DECLINING BALANCE METHOD
449 REM - ENTER DECLINING BALANCE PERCENT
450 PRINT "METHOD IN %";
460 INPUT M
469 REM - CONVERT PERCENT TO DECIMAL
470 M=M/100
479 REM - N COUNTS THE LINES PRINTED ON EACH PAGE
489 REM - R IS THE AMOUNT TO DEPRECIATE
490 R=D
500 PRINT "□";
570 PRINT TAB(8);"DECLINING BALANCE ";
571 PRINT "TAX"
575 PRINT TAB(12);"DEPRECIATION"
580 PRINT TAB(14);"PRICE $";V
590 PRINT TAB(7);"SALVAGE VALUE $";S
600 PRINT TAB(5);"NET DEPRECIATED $";
```

```
601 PRINT V-S
610 PRINT TAB(13);"LIFE";Y;"YEARS"
620 PRINT TAB(13);"METHOD ";M*100;"%"
630 PRINT
640 PRINT "YEAR   DEPRECIATION";
641 PRINT "  BALANCE"
642 PRINT
643 N=11
645 FOR I=1 TO Y
649 REM - COMPUTE DEPRECIATION AND ROUND OFF TO THE NEAREST CENT
650 R1=INT((R*M/Y)*100+.5)/100
659 REM - ACCUMULATE REMAINING BALANCE
660 R=R-R1
670 PRINT I;"    ";R1,R
675 N=N+1
677 REM - CHECK FOR FULL SCREEN (23 LINES)
680 IF N/23=INT(N/23) THEN GOSUB 800
690 NEXT I
700 PRINT
709 REM - RESTART OR END PROGRAM?
710 PRINT "MORE DATA? (1=YES, 0=NO)";
720 INPUT X
730 IF X=1 THEN 20
740 END
```

Check Writer

This program prints a check. You must provide the date, amount and payee of the check. The program translates the date and amount to words and prints providing spacing within the check.

You should regard the program listed below as a sample of a check-writing program. Very few checks will conform exactly to the spacing provided in this program. The method of translating words from numbers is generally applicable. Spacing should be altered to conform to your own check format.

File commands to open and close the flow of data to the printer are located at statement numbers 250 and 625 respectively. The PRINT statements between 250 and 625 do not need to be replaced with PRINT# when using the CMD command at 250.

When the program asks the question READY TO PRINT CHECK? it is prompting you to insert a blank check in your printing device. The check should be set one line above the line on which the date is to be printed.

Once the check is set up, key RETURN (no other entry is required) and the check will be printed.

Example:

Among the checks that Miracle Corporation must write are one to Osborne & Associates for $4975.89 and one to Freida Alexander for $103.75. Print the checks using the computer.

```
CHECK WRITER

DATE (MMDDYY)? 30780
-----(TO END PROGRAM ENTER 'END')-----
FIRST NAME OF PAYEE? OSBORNE/

LAST NAME OF PAYEE? MCGRAW-HILL

AMOUNT OF CHECK? 4975.89

READY TO PRINT CHECK?
```

HEAVENLY BANK
EMERYVILLE OFFICE
4120 ASHBY AVENUE
EMERYVILLE, CA 94601

NO. 328

MARCH 7 19 80

$ 4975.89

AMOUNT $ 4975.89

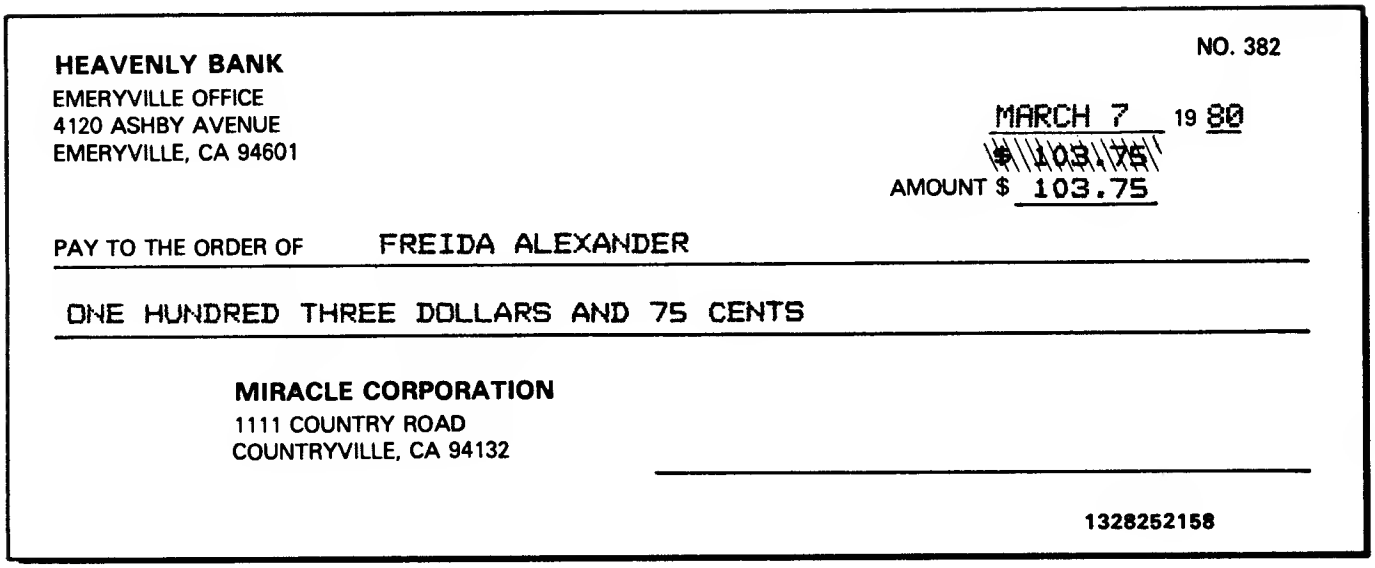
PAY TO THE ORDER OF OSBORNE/MCGRAW-HILL

FOUR THOUSAND NINE HUNDRED SEVENTY-FIVE DOLLARS AND 89 CENTS

MIRACLE CORPORATION
1111 COUNTRY ROAD
COUNTRYVILLE, CA 94132

1328252158

```
FIRST NAME OF PAYEE? FREIDA

LAST NAME OF PAYEE? ALEXANDER

AMOUNT OF CHECK? 103.75

READY TO PRINT CHECK?
```



HEAVENLY BANK
EMERYVILLE OFFICE
4120 ASHBY AVENUE
EMERYVILLE, CA 94601

NO. 382

MARCH 7 19 80

$ 103.75

AMOUNT $ 103.75

PAY TO THE ORDER OF FREIDA ALEXANDER

ONE HUNDRED THREE DOLLARS AND 75 CENTS

MIRACLE CORPORATION
1111 COUNTRY ROAD
COUNTRYVILLE, CA 94132

1328252158

```
FIRST NAME OF PAYEE? END
```

```
10 PRINT "□CHECK WRITER"
20 PRINT
30 DATA "ONE","TWO","THREE","FOUR"
31 DATA "FIVE","SIX","SEVEN","EIGHT"
40 DATA "NINE","TEN","ELEVEN","TWELVE"
41 DATA "THIRTEEN","FOURTEEN"
42 DATA "FIFTEEN","SIXTEEN"
50 DATA "SEVENTEEN","EIGHTEEN"
51 DATA "NINETEEN","TWENTY","THIRTY"
60 DATA "FORTY","FIFTY","SIXTY"
61 DATA "SEVENTY","EIGHTY","NINETY"
70 DATA "JANUARY","FEBRUARY","MARCH"
71 DATA "APRIL","MAY","JUNE","JULY"
80 DATA "AUGUST","SEPTEMBER","OCTOBER"
81 DATA "NOVEMBER","DECEMBER"
89 REM - ENTER DATE WITHOUT COMMAS; DAY AND YEAR MUST CONTAIN TWO DIGITS
90 PRINT "DATE (MMDDYY)";
100 INPUT D
110 PRINT "-----(TO END PROGRAM ";
111 PRINT "ENTER 'END')-----"
120 PRINT "FIRST NAME OF PAYEE";
130 INPUT F$:PRINT
139 REM - END PROGRAM?
140 IF F$="END" THEN 790
150 PRINT "LAST NAME OF PAYEE";
160 INPUT L$:PRINT
170 PRINT "AMOUNT OF CHECK";
```

```
180 INPUT A:PRINT
189 REM - INSERT BLANK CHECK IN PRINTING DEVICE, KEY <RETURN> WHEN READY
190 PRINT "READY TO PRINT CHECK?";
200 GET Z$: IF Z$="" GOTO 200
201 PRINT
202 PRINT
203 PRINT
204 PRINT
209 REM - BREAK ENTERED DATE NUMBER INTO MONTH, DAY, YEAR FIGURES
210 D1=INT(D/10000)
220 D2=INT((D-D1*10000)/100)
230 D3 = INT(D-(D1*100+D2)*100)
235 D3=D3+1900
239 REM - GOTO CORRECT MONTH IN DATA TABLE
240 RESTORE
241 FOR K= 1 TO 27+D1
242 READ X$
243 NEXT K
250 OPEN4,4:CMD4
259 REM - PRINT DATE
260 PRINT TAB(20);X$;D2;",";D3
269 REM - PRINT AMOUNT TWICE; FIRST TIME FOR SHADED BOX
270 PRINT TAB(24);"$";A
280 PRINT TAB(25);A
290 PRINT
300 PRINT F$;" ";L$
310 PRINT
319 REM - AMOUNT OF CHECK LEGITIMATE?
320 IF A<=0 THEN 770
330 A1=A
339 REM - AMOUNT IN THE THOUSANDS?
340 N1=INT(A1/1E3)
349 REM - CAN'T PRINT FOR AMOUNT OVER $99999.99
350 IF N1>99 THEN 770
360 IF N1=0 THEN 390
370 GOSUB 640
380 PRINT "THOUSAND ";
390 A1=A1-N1*1E3
399 REM - AMOUNT IN THE HUNDREDS?
400 N1=INT(A1/100)
410 IF N1=0 THEN 440
420 GOSUB 640
430 PRINT "HUNDRED ";
440 A1=A1-N1*100
449 REM - AMOUNT IN THE ONES OR TENS?
450 N1=INT(A1)
460 IF N1>0 THEN 490
470 IF A>=1 THEN 500
480 GOTO 510
490 GOSUB 640
500 PRINT "DOLLARS ";
510 A1=A1-N1
519 REM - ANY CENTS?
520 IF A1<.01 THEN 600
529 REM - IF AMOUNT IS CENTS ONLY; DON'T PRINT 'AND'
530 IF A<1 THEN 550
540 PRINT "AND";
```

```
550 A1=INT(A1*100)+.5/100
559 REM - CENTS ARE PRINTED IN NUMERIC FORM
560 PRINT INT(A1);"CENTS"
569 REM - SPACE OFF OF CHECK
570 PRINT
580 PRINT
590 PRINT
600 PRINT
610 PRINT
620 PRINT
625 PRINT#4:CLOSE4
629 REM - RESTART PROGRAM
630 GOTO 120
639 REM - SUBROUTINE TO GET WORDS FOR NUMBERS
640 IF N1<21 THEN 730
650 RESTORE
651 FOR K=1 TO (N1-20)/10+20
652 READ X$
653 NEXT K
670 PRINT X$;
680 A3=N1-INT(N1/10)*10
690 IF A3=0 THEN 760
700 PRINT "-";
710 RESTORE
711 FOR K=1 TO A3
712 READ X$
713 NEXT K
720 GOTO 750
730 RESTORE
731 FOR K=1 TO N1
732 READ X$
733 NEXT K
750 PRINT X$;" ";
759 REM - END OF SUBROUTINE
760 RETURN
770 PRINT,"*****VOID*****"
780 GOTO 570
790 END
```

Recipe Cost

This program calculates the cost and the cost per serving of a single recipe. For each ingredient you must provide the purchase price, the amount purchased, the amount used in the recipe, and the number of recipe units per purchase unit.

Example:

Listed below is a recipe for strawberry shortcake. Calculate the cost of the recipe and the cost per serving. What would the cost per serving be if one cake serves 12? The conversion factors and price per ingredient are supplied.

Strawberry Shortcake — 8 servings

3 c. flour	2.5 c./lb.	$1.59	5 lb.
3¼ tsp. baking powder	15 tsp./oz.	.43	4 oz.
¼ c. sugar	2 c./lb.	1.24	5 lb.
1¼ tsp. salt	6 tsp./oz.	.29	1 lb.
½ c. butter	2 c./lb.	1.49	1 lb.
1 egg	12/doz.	.75	1 doz.
⅔ c. milk	4 c./qt.	.40	1 qt.
3 pts strawberries	——	.49	1 pt.
½ pt. whipping cream	——	.59	½ pt.

```
RECIPE COST

NUMBER OF INGREDIENTS? 9
INGREDIENT 1 :
STORE COST FOR BULK UNIT? 1.59
NUMBER OF UNITS IN BULK? 5
RECIPE UNITS PER BULK UNIT? 2.5
NUMBER OF RECIPE UNITS NEEDED? 3

INGREDIENT 2 :
STORE COST FOR BULK UNIT? .43
NUMBER OF UNITS IN BULK? 4
RECIPE UNITS PER BULK UNIT? 15
NUMBER OF RECIPE UNITS NEEDED? 3.25

INGREDIENT 3 :
STORE COST FOR BULK UNIT? 1.24
NUMBER OF UNITS IN BULK? 5
RECIPE UNITS PER BULK UNIT? 2
NUMBER OF RECIPE UNITS NEEDED? .25

INGREDIENT 4 :
STORE COST FOR BULK UNIT? .29
NUMBER OF UNITS IN BULK? 1
RECIPE UNITS PER BULK UNIT? 96
NUMBER OF RECIPE UNITS NEEDED? 1.25
```

```
INGREDIENT 5 :
STORE COST FOR BULK UNIT? 1.49
NUMBER OF UNITS IN BULK? 1
RECIPE UNITS PER BULK UNIT? 26
NUMBER OF RECIPE UNITS NEEDED? .5

INGREDIENT 6 :
STORE COST FOR BULK UNIT? .75
NUMBER OF UNITS IN BULK? 1
RECIPE UNITS PER BULK UNIT? 12
NUMBER OF RECIPE UNITS NEEDED? 1

INGREDIENT 7 :
STORE COST FOR BULK UNIT? .40
NUMBER OF UNITS IN BULK? 1
RECIPE UNITS PER BULK UNIT? 4
NUMBER OF RECIPE UNITS NEEDED? .6666667

INGREDIENT 8 :
STORE COST FOR BULK UNIT? .49
NUMBER OF UNITS IN BULK? 1
RECIPE UNITS PER BULK UNIT? 1
NUMBER OF RECIPE UNITS NEEDED? 3

INGREDIENT 9 :
STORE COST FOR BULK UNIT? .59
NUMBER OF UNITS IN BULK? 1
RECIPE UNITS PER BULK UNIT? 1
NUMBER OF RECIPE UNITS NEEDED? 1

NUMBER OF SERVINGS? 8

TOTAL COST FOR ONE RECIPE = $ 3
COST PER SERVING = $ .38

MORE OR FEWER SERVINGS (1=YES, 0=NO)? 1
NUMBER OF SERVINGS? 12

TOTAL COST FOR ONE RECIPE = $ 3
COST PER SERVING = $ .25

MORE OR FEWER SERVINGS (1=YES, 0=NO)? 0
```

```
10 PRINT "RECIPE COST"
15 PRINT
19 REM - STATMENTS 30 TO 180 REQUEST USER INPUT
20 PRINT "NUMBER OF INGREDIENTS";
30 INPUT N
39 REM - LOOP TO REQUEST DATA FOR EACH INGREDIENT
40 FOR I=1 TO N
50 PRINT
60 PRINT "INGREDIENT";I;":"
70 PRINT "STORE COST FOR BULK UNIT";
80 INPUT C
90 PRINT "NUMBER OF UNITS IN BULK";
100 INPUT U
```

```
110 PRINT "RECIPE UNITS PER ";
111 PRINT "BULK UNIT";
120 INPUT F
130 PRINT "NUMBER OF RECIPE ";
131 PRINT "UNITS NEEDED";
140 INPUT R
149 REM - SUM COST OF EACH INGREDIENT PER AMOUNT USED
150 P=P+C/U/F*R
160 NEXT I
165 PRINT
170 PRINT "NUMBER OF SERVINGS";
180 INPUT S
190 PRINT
199 REM - ROUND OFF COSTS TO NEAREST CENT, PRINT RESULTS
200 PRINT "TOTAL COST FOR ";
201 PRINT "ONE RECIPE = $";
205 PRINT INT(P*100+.5)/100
210 PRINT "COST PER SERVING = $";
215 PRINT INT(P/S*100+.5)/100
220 PRINT
229 REM - CALCULATE ALTERNATIVE PRICE PER SERVING
230 PRINT "MORE OR FEWER SERVINGS ";
231 PRINT "(1=YES, 0=NO)";
240 INPUT N
250 IF N=1 THEN 170
260 END
```

OPTION

As you become familiar with the operation of this program you may wish to shorten it by entering the information required for each ingredient on one line. The program changes necessary are listed following the example below.

Example:

Calculate the cost per serving of Strawberry Shortcake in the previous example when it is served without cream.

```
RECIPE COST

NUMBER OF INGREDIENTS? 8
INGREDIENT 1 ? 1.59,5,2.5,3
INGREDIENT 2 ? .43,4,15,3.25
INGREDIENT 3 ? 1.24,5,2,.25
INGREDIENT 4 ? .29,1,96,1.25
INGREDIENT 5 ? 1.49,1,2,.5
INGREDIENT 6 ? .75,1,12,1
INGREDIENT 7 ? .40,1,4,.6666667
INGREDIENT 8 ? .49,1,1,3

NUMBER OF SERVINGS? 8

TOTAL COST FOR ONE RECIPE = $ 2.41
COST PER SERVING = $ .3

MORE OR FEWER SERVINGS (1=YES, 0=NO)? 1
NUMBER OF SERVINGS? 12
```

```
TOTAL COST FOR ONE RECIPE = $ 2.41
COST PER SERVING = $ .2

MORE OR FEWER SERVINGS (1=YES, 0=NO)? 0
```

```
1 REM - OPTION 55-70
10 PRINT "□RECIPE COST"
 .
 .
 .
40 FOR I=1 T0 N
50 PRINT
55 REM - ENTER C,U,F,R
56 REM - WHERE C=COST FOR BULK UNIT
57 REM -       U=NUMBER UNITS IN BULK UNIT
58 REM -       F=RECIPE UNITS PER BULK UNIT
59 REM -       R=NUMBER RECIPE UNITS CALLED FOR
60 PRINT "INGREDIENT";I;
70 INPUT C,U,F,R
149 REM - SUM COST OF EACH INGREDIENT PER AMOUNT USED
 .
 .
 .
260 END
```

Survey Check (Map Check)

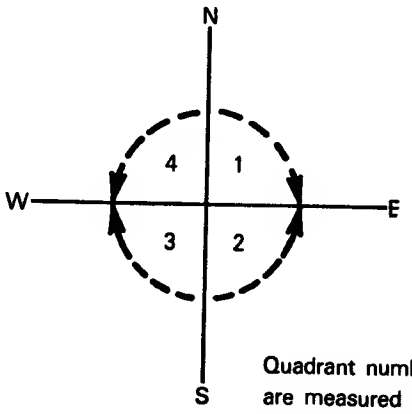
Courtesy: Robert Irving
Northridge,
California

This program calculates the error of closure and area of a plot for which a traverse of the perimeter is available. The program will also calculate how far North and East the end of an open traverse is from its origin (the Northing and Easting). The local coordinates of the origin can be entered for an open traverse. Negative values of Northing and Easting are South and West, respectively, of the 0,0 origin of the survey.

The individual legs of the traverse may be either straight lines or arcs of circles. To compute the traverse, you must have the bearing and length of each straight leg. You also need the radius, bearing of chord, and length of chord (or radius, arc measure, and bearing of a tangent) for each curved leg.

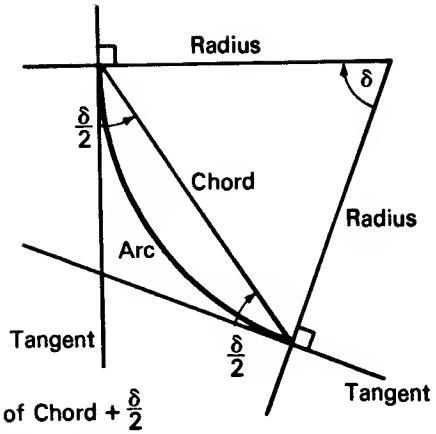
For a closed survey, pick any intersection of legs as a starting point, and number the lines and arcs, starting with one, in a *clockwise* direction around the perimeter. If any arc is 180 degrees or more, it must be broken into smaller arcs, each less than 180 degrees.

By convention, surveyors measure bearings East and West of North and South, as shown in the following figure. This convention was established in the days before computers, so that trigonometric functions could be easily looked up in tables not exceeding 90 degrees. For each leg, you must enter the quadrant number and the degrees, minutes and seconds East or West of the North-South axis. The program will indicate the direction of the leg (e.g., SW), and will convert the quadrant, degrees, etc. to an azimuth angle. Azimuth is measured clockwise from North to 360 degrees.



Quadrant numbering scheme. Note angles are measured from the North-South line.

A curved leg, or arc, is defined by two auxiliary legs, each of which is a radius of the arc. The bearing of the first auxiliary leg is the direction of the radius from the first encountered end of the arc to the center of the arc. You can compute this bearing from the bearing of the arc's tangent at that point, since the radius is perpendicular to the tangent. The survey may show the bearing of the tangent. If not, you can compute it by adding one half the angular extent of the arc to the bearing of the arc's chord, as shown in the next figure.



Bearing of Tangent = Bearing of Chord + $\frac{\delta}{2}$

The bearing of the second radius is from the center of the arc to the other end, and the distance is entered as a *negative* number to signal to the computer that this and the prior leg are not perimeter legs, but auxiliary legs of an arc.

The program asks you for the bearing and distance of each leg by number. Legs are entered in sets of ten (or less). Following the last entry in a set, you can correct any leg in the set. You must enter both auxiliary legs of an arc in the same set. You can enter a bearing of zero to end one set, and then enter more legs on the next set.

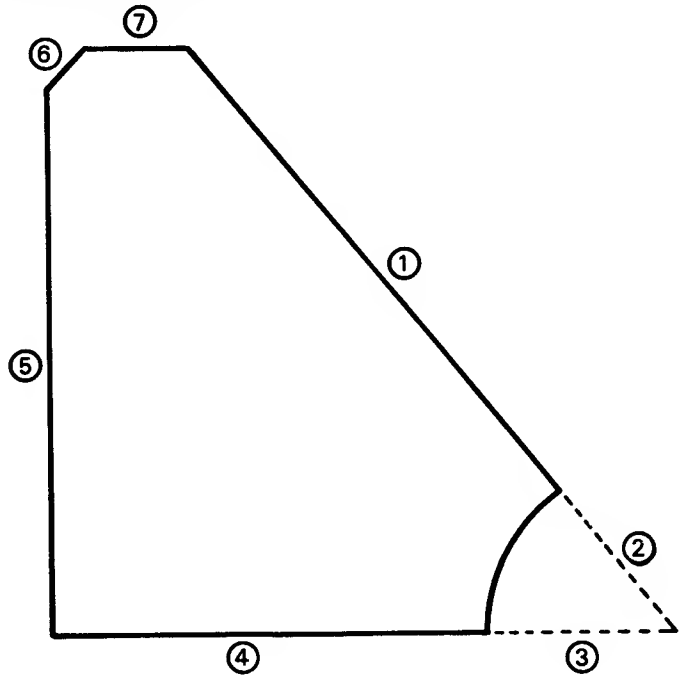
When you have corrected a set, a traverse table is printed for the set. This includes each leg number, direction, azimuth angle and distance, and incremental and cumulative Northing and Easting. The cumulative Northing and Easting after the last leg on a closed survey gives the error of closure. Arc angle, radius, sector area, chord length, and tangent length are printed between the two auxiliary legs of each curved leg.

Following the printout of the last leg of a closed survey, the area of the plot will be printed, both in square feet and in acres. The area computed is very accurate provided two conditions are met:

1) the error of closure is small (0.01 feet is usual for a house lot), and
2) the area is sufficiently small that curvature of the earth does not become significant. Surveys covering several tens of miles have to account for this latter factor.

Example:

The figure below illustrates the boundaries of a lot with one curved side. The leg numbers are circled. Bearings and distances are shown for each leg. Find the error of closure and lot area.



① S39°0"E
149.83
② S39°0"E
50.00
③ N85°23'53"W
50.00
④ N85°23'53"W
114.32
⑤ N1°5'0"E
132.78
⑥ N46°0'0"E
14.00
⑦ S89°0'0"E
25.46

```
 MAP CHECK

 OPEN (1) OR CLOSED (0) SURVEY? 0

 NEXT SET OF  LEGS:

 LEG NO.  1
 QUADRANT,DEGREES,MINUTES,SECONDS
 ? 2,39,0,0
 DISTANCE
 (NEGATIVE IF OUTWARD RADIUS)? 149.83

 LEG NO.  2
 QUADRANT,DEGREES,MINUTES,SECONDS
 ? 2,39,0,0
 DISTANCE
 (NEGATIVE IF OUTWARD RADIUS)? 50

 LEG NO.  3
 QUADRANT,DEGREES,MINUTES,SECONDS
 ? 4,85,23,53
 DISTANCE
 (NEGATIVE IF OUTWARD RADIUS)? -50
 LEG NO.  4
 QUADRANT,DEGREES,MINUTES,SECONDS
 ? 4,85,23,53
 DISTANCE
 (NEGATIVE IF OUTWARD RADIUS)? 114.32

 LEG NO.  5
 QUADRANT,DEGREES,MINUTES,SECONDS
 ? 1,1,5,0
 DISTANCE
 (NEGATIVE IF OUTWARD RADIUS)? 132.78

 LEG NO.  6
 QUADRANT,DEGREES,MINUTES,SECONDS
 ? 1,46,0,0
 DISTANCE
 (NEGATIVE IF OUTWARD RADIUS)? 14

 LEG NO.  7
 QUADRANT,DEGREES,MINUTES,SECONDS
 ? 2,89,0,0
 DISTANCE
 (NEGATIVE IF OUTWARD RADIUS)? 25.46

 LEG NO.  8
 QUADRANT,DEGREES,MINUTES,SECONDS
 ? 0,0,0,0
 CORRECT WHICH LEG IN THIS SET?
        (0=NO MORE CHANGES)? 0
```

```
ORIGIN   0 / 0

LEG/DIR.              1 /SE
AZIMUTH/DIST.         141  0  0 / 149.83
DEL N/DEL E          -116.44 / 94.291
NORTHING/EASTING     -116.44 / 94.291

LEG/DIR.              2 /SE
AZIMUTH/DIST.         141  0  0 / 50
DEL N/DEL E          -38.857 / 31.466
NORTHING/EASTING     -155.297 / 125.757

 ARC: 46  23  53
   R= 50 A= 2024.497 C= 39.393 T= 21.429

LEG/DIR.              3 /NW
AZIMUTH/DIST.         274  36  7 / 50
DEL N/DEL E           4.012 /-49.839
NORTHING/EASTING     -151.285 / 75.918

** PRESS 'C' FOR NEXT SET OF LEGS **

READY.
1:177C1:177C

LEG/DIR.              4 /NW
AZIMUTH/DIST.         274  36  7 / 114.32
DEL N/DEL E           9.172 /-113.951
NORTHING/EASTING     -142.113 /-38.033

LEG/DIR.              5 /NE
AZIMUTH/DIST.         1  5  0 / 132.78
DEL N/DEL E           132.756 / 2.51
NORTHING/EASTING     -9.357 /-35.523

LEG/DIR.              6 /NE
AZIMUTH/DIST.         46  0  0 / 14
DEL N/DEL E           9.725 / 10.071
NORTHING/EASTING       .368 /-25.452

** PRESS 'C' FOR NEXT SET OF LEGS **

READY.

LEG/DIR.              7 /SE
AZIMUTH/DIST.         91  0  0 / 25.46
DEL N/DEL E          -.444 / 25.456
NORTHING/EASTING     -.076 / 4E-03

ANY MORE LEGS (1=YES, 0=NO)? 0
PLOT AREA IS 13347.683 SQ. FT.

PLOT AREA IS .30642064 ACRES

BREAK IN  1000
```

```
 10 REM - SURVEY CHECK
 12 REM - FOR CLOSED SURVEY FOLLOW TRAVERSE CLOCKWISE
 13 REM - KEEP PLOT TO RIGHT OF EACH PERIMETER LEG
 14 REM - COMPUTE AUXILIARY LEGS AS RADII AT EACH
 15 REM - END OF ARC.  ARC<180 DEGREES
 16 REM
 17 REM - VALUE OF PI; PI=3.14159265
 19 REM - K0 = NUMBER OF LEGS PER SET
 20 K0=10
 30 DIM B(10),L(10)
 39 REM - FNR(X) ROUNDS X TO 3 DECIMAL PLACES
 40 DEF FNR(X)=INT(X*1000+.5)/1000
 49 REM - R IS THE CONVERSION FACTOR FOR DEGREES TO RADIANS
 50 R=1.745329251E-2
 60 PRINT "□MAP CHECK":PRINT
 70 PRINT "OPEN (1) OR CLOSED (0) SURVEY";
 80 INPUT F
 90 IF F=0 THEN 120
100 PRINT "ORIGIN: NORTHING, EASTING";
110 INPUT N,E
120 PRINT:PRINT "NEXT SET OF  LEGS:":PRINT
125 G=H
130 FOR K=1 TO K0
139 REM - INPUT BEARING AND DISTANCE FOR NEXT LEG
140 GOSUB 2000
149 REM - IF BEARING IS 0, END INPUT FOR THIS SET
150 IF Q=0 THEN 170
155 G=G+1
160 GOTO 240
169 REM - ZERO UNUSED LEGS IN THIS SET
170 IF K=K0 THEN 230
180 FOR J=K+1 TO K0
190 B(J)=0
200 L(J)=0
210 NEXT J
230 K=K0
240 NEXT K
260 PRINT "CORRECT WHICH LEG IN THIS SET"
261 PRINT "       (0=NO MORE CHANGES)";
270 INPUT K
279 REM - NO CHANGES IF 0 INPUT
280 IF K=0 THEN 310
285 K=K-H
290 GOSUB 2000
300 GOTO 260
309 REM - COMPUTE VALUES AND PRINT TRAVERSE TABLE
310 PRINT "□ORIGIN  ";FNR(N);"/";FNR(E)
340 PRINT
350 FOR K=1 TO K0
360 L1=L(K)
361 Z=Z+1
369 REM - CHECK FOR ARC
370 IF L1<0 THEN 1100
380 IF L1=0 THEN 900
388 REM - COMPUTE NORTHING/EASTING INCREMENT (CONVERT BEARINGS FROM
389 REM - DEGREES TO RADIANS)
390 L=L(K)*COS(B(K)*R)
```

```
400 D=L(K)*SIN(B(K)*R)
410 N=N+L
420 E=E+D
429 REM - INCREMENT AREA
430 A=A-E*L+N*D
440 PRINT "LEG/DIR.";TAB(17);H+K;"/";
449 REM - FROM BEARING, DETERMINE DIRECTION
450 IF B(K)=0 THEN 470
460 GOTO 490
470 PRINT "N"
480 GOTO 830
490 IF B(K)<90 THEN 510
500 GOTO 530
510 PRINT "NE"
520 GOTO 830
530 IF B(K)=90 THEN 550
540 GOTO 570
550 PRINT "E"
560 GOTO 830
570 IF B(K)<180 THEN 590
580 GOTO 610
590 PRINT "SE"
600 GOTO 830
610 IF B(K)=180 THEN 630
620 GOTO 650
630 PRINT "S"
640 GOTO 830
650 IF B(K)<270 THEN 670
660 GOTO 690
670 PRINT "SW"
680 GOTO 830
690 IF B(K)=270 THEN 710
700 GOTO 730
710 PRINT "W"
720 GOTO 830
730 IF B(K)<360 THEN 750
740 GOTO 770
750 PRINT "NW"
760 GOTO 830
770 IF B(K)=360 THEN 790
780 GOTO 810
790 PRINT "N"
800 GOTO 830
810 B(K)=B(K)-360
820 GOTO 450
829 REM - BREAK BEARING INTO DEGREES, MINUTES, SECONDS
830 D1=INT(B(K))
840 M1=(B(K)-D1)*60
850 M=INT(M1)
860 S=INT((M1-M)*60+.5)
870 PRINT "AZIMUTH/DIST.     ";D1;M;S;"/";FNR(L(K))
880 PRINT "DEL N/DEL E       ";FNR(L);"/";FNR(D)
881 PRINT "NORTHING/EASTING ";FNR(N);"/";FNR(E)
885 PRINT
890 L(K)=L1
891 IF Z<3 THEN 900
895 PRINT "** PRESS 'C' FOR NEXT SET ";      "OF LEGS **"
```

```
896 GET W$:IF W$="" THEN 896
897 PRINT
898 Z=0
900 NEXT K
910 H=G
920 PRINT "ANY MORE LEGS (1=YES, 0=NO)";
930 INPUT U
940 IF U<>0 THEN THEN 120
949 REM - NO AREA FOR OPEN SURVEY
950 IF F<>0 THEN 1000
960 A=ABS(A/2)
970 PRINT "PLOT AREA IS";FNR(A);"SQ. FT."
980 PRINT
990 PRINT "PLOT AREA IS";INT(A/43560*1E8+.5)/1E8;"ACRES"
1000 STOP
1099 REM - CALCULATE CURVED LEG AND PRINT ON TRANSVERSE TABLE
1100 C=ABS(B(K)-B(K-1))
1110 C=ABS(180-C)
1120 D=-L1
1130 L(K)=D
1140 A1=C/180*π*D*D
1150 C1=2*D*SIN(C/2*R)
1160 T=D*TAN(C/2*R)
1170 B9=B(K)-B(K-1)
1180 IF B9<-180 THEN 1210
1190 IF B9>180 THEN 1210
1200 IF B9>0 THEN 1230
1210 A=A+A1
1220 GOTO 1240
1230 A=A-A1
1240 D1=INT(C)
1250 M1=(C-D1)*60
1260 M=INT(M1)
1270 S=INT((M1-M)*60+.5)
1280 PRINT " ARC:"D1;M;S:PRINT"    R=";FNR(D);"A=";FNR(A1);"C=";
1290 PRINT FNR(C1);"T=";FNR(T)
1300 PRINT
1320 GOTO390
1999 REM - INPUT DATA FOR ONE LEG
2000 B(K)=0
2010 L(K)=0
2020 PRINT "LEG NO. ";H+K;"
2021 PRINT "QUADRANT,DEGREES,";
2022 PRINT "MINUTES,SECONDS":
2030 INPUT Q,D,M,S
2040 IF Q=0 THEN 2270
2050 IF Q>4 THEN 2020
2060 IF Q<0 THEN 2020
2070 IF D<0 THEN 2020
2080 IF M<0 THEN 2020
2090 IF S<0 THEN 2020
2100 B(K)=D+(M+S/60)/60
2110 IF B(K)>90 THEN 2020
2120 IF Q=1 THEN 2220
2130 IF Q=2 THEN 2150
2140 GOTO 2170
2150 B(K)=180-B(K)
```

```
2160 GOTO 2220
2170 IF Q=3 THEN 2190
2180 GOTO 2210
2190 B(K)=180+B(K)
2200 GOTO 2220
2210 IF Q<>4 THEN 2220
2215 B(K)=360-B(K)
2220 PRINT "DISTANCE "
2221 PRINT "(NEGATIVE IF OUTWARD RADIUS)";
2230 INPUT L(K):PRINT
2240 IF L(K)>0 THEN 2270
2250 IF ABS(L(K))<>ABS(L(K-1))THEN 2220
2270 RETURN
9999 END
```

Day of the Week

This program calculates the day of the week that a given date falls on. It will figure, for example, that December 25, 1980 will be a Thursday.

You must enter the date in numeric form and in the order of month, day, year. September 12, 1975 will be entered as `9,12,1975`, making certain that commas, not slashes or dashes, separate the figures.

Examples:

Cindy's birthdate is March 4, 1953. On what day was she born?

Uncle Lon has an appointment on September 30, 1977. What day is that on?

```
DAY OF THE WEEK

(ENTER 0,0,0 TO END PROGRAM)
MONTH,DAY,YEAR? 3,4,1953
WEDNESDAY

MONTH,DAY,YEAR? 9,30,1977
FRIDAY

MONTH,DAY,YEAR? 0,0,0
```

```
10 PRINT "□DAY OF THE WEEK"
20 PRINT
29 REM - REQUEST USER INPUT
30 PRINT "(ENTER 0,0,0 TO END PROGRAM)"
40 PRINT "MONTH,DAY,YEAR";
50 INPUT M,D,Y
59 REM - TEST FOR END OF PROGRAM
60 IF M<>0 THEN 100
70 IF D<>0 THEN 100
80 IF Y<>0 THEN 100
90 GOTO 360
99 REM - NEED TO ADJUST INPUT FOR CALCULATIONS?
100 IF M>2 THEN 130
109 REM - ADJUST INPUT
110 M=M+12
120 Y=Y-1
129 REM - CALCULATE DAY NUMBER
130 N=D+2*M+INT(.6*(M+1))+Y+INT(Y/4)-INT(Y/100)+INT(Y/400)+2
140 N=INT((N/7-INT(N/7))*7+.5)
149 REM - FIND CORRECT DAY NUMBER, TRANSLATE TO DAY, PRINT
150 IF N>0 THEN 180
160 PRINT "SATURDAY"
170 GOTO 340
180 IF N>1 THEN 210
190 PRINT "SUNDAY"
200 GOTO 340
```

```
210 IF N>2 THEN 240
220 PRINT "MONDAY"
230 GOTO 340
240 IF N>3 THEN 270
250 PRINT "TUESDAY"
260 GOTO 340
270 IF N>4 THEN 300
280 PRINT "WEDNESDAY"
290 GOTO 340
300 IF N>5 THEN 330
310 PRINT "THURSDAY"
320 GOTO 340
330 PRINT "FRIDAY"
340 PRINT
349 REM - RESTART PROGRAM
350 GOTO 40
360 END
```

Days between Two Dates

This program calculates the number of days between two given dates. Leap years are taken into account. The program assumes there is one day between today and tomorrow. For instance, there are two days between March 1 and March 3 of the same year.

There are a few precautions to assure the proper use of this program. First, you must be certain to enter the earlier date first. Second, dates must be entered in number form (**3**, not **MARCH**) and in the correct order (month, day, year, i.e., **3,17,1976**). Commas, not slashes or dashes, must separate the figures. Third, the year must not be abbreviated (**1976** , not **76**), even if both dates are in the same century. Finally, the month entered must not be greater than 12 and the days no greater than the number of days in the particular month. If such is the case, the message **UNREAL DATE** is printed to alert you to the fact that an unreal date (such as **14,32,1975**) has been entered. An incorrect answer is likely to result.

Example:

John's birthdate is August 8, 1951. How many days old will he be on his 30th birthday?

```
DAYS BETWEEN TWO DATES

FIRST DATE? 7,4,1977
SECOND DATE? 12,25,1977
DIFFERENCE = 174 DAYS

MORE DATA (1=YES, 0=NO)? 0
```

```
10 PRINT "□DAYS BETWEEN TWO DATES"
20 PRINT
29 REM - STATEMENTS 30 TO 60 REQUEST USER INPUT
30 PRINT "FIRST DATE";
40 INPUT M1,D1,Y1
50 PRINT "SECOND DATE";
60 INPUT M2,D2,Y2
69 REM - SET VARIABLES TO BE USED IN SUBROUTINE
70 M=M1
80 D=D1
90 Y=Y1
100 GOSUB 230
109 REM - SAVE COMPUTED NUMBER OF DAYS IN N
110 N=A
119 REM - SET VARIABLES TO BE USED IN SUBROUTINE
120 M=M2
130 D=D2
140 Y=Y2
150 GOSUB 230
159 REM - CALCULATE DIFFERENCE AND PRINT
160 N=A-N
170 PRINT "DIFFERENCE =";N;"DAYS"
180 PRINT
189 REM - RESTART OR END PROGRAM? USER INPUT REQUIRED
```

```
190 PRINT "MORE DATA (1=YES, 0=NO)";
200 INPUT X
210 IF X=1 THEN 20
219 REM - END PROGRAM
220 GOTO 460
227 REM - SUBROUTINE TO COMPUTE NUMBER OF DAYS FROM 0/0/0/ TO M/D/Y
228 REM - START WITH TEST FOR UNREAL DATE
229 REM- GO TO CORRECT TEST DEPENDING ON NUMBER OF DAYS IN THE MONTH
230 ON M GOTO 260,280,260,340,260,340,260,260,340,260,340,260
240 PRINT "UNREAL DATE"
249 REM - STOP CALCULATIONS, RETURN TO MAIN PROGRAM
250 RETURN
259 REM - MONTH HAS 31 DAYS
260 IF D>31 THEN 240
270 GOTO 350
279 REM - MONTH IS FEBRUARY; A LEAP YEAR?
280 IF Y/4<>INT(Y/4) THEN 310
290 IF Y/400=INT(Y/400) THEN 320
300 IF Y/100<>INT(Y/100) THEN 320
309 REM - NOT A LEAP YEAR; MONTH HAS 28 DAYS
310 IF D>28 THEN 240
319 REM - A LEAP YEAR; MONTH HAS 29 DAYS
320 IF D>29 THEN 240
330 GOTO 350
339 REM - MONTH HAS 30 DAYS
340 IF D>30 THEN 240
349 REM - TABLE OF NUMBER OF DAYS FROM FIRST OF YEAR TO FIRST OF EACH
    MONTH
350 DATA 0,31,59,90,120,151,181,212
351 DATA 243,273,304,334
360 RESTORE
361 FOR Q=1 TO M
362 READ A
363 NEXT Q
369 REM - GET NUMBER OF DAYS FROM JANUARY 1 TO FIRST OF MONTH FROM DATA
    TABLE
379 REM - COMPUTE NUMBER OF DAYS FROM 0/0/0 TO M/D/Y
380 A=A+Y*365+INT(Y/4)+D+1-INT(Y/100)+INT(Y/400)
389 REM - POSSIBLY A LEAP YEAR?
390 IF INT(Y/4)<>Y/4 THEN 450
409 REM - CONTINUE TEST FOR LEAP YEAR
410 IF Y/400= INT(Y/400) THEN 430
420 IF Y/100=INT(Y/100) THEN 440
428 REM - IF MONTH IS JANUARY OR FEBUARY, ADJUST CALCLATED NUMBER OF
    DAYS
430 IF M>2 THEN 450
440 A=A-1
449 REM - END OF SUBROUTINE, RETURN TO MAIN PROGRAM
450 RETURN
460 END
```

OPTION

To shorten this program you may wish to omit the test for unreal dates. It should be noted that if a month of more than 12 is entered when this test is omitted, an input error will result. The program lines which may be deleted are listed following the example below.

Example:

How many days are there between July 4 and Christmas?

```
DAYS BETWEEN TWO DATES

FIRST DATE? 8,8,1951
SECOND DATE? 8,8,1981
DIFFERENCE = 10958 DAYS

MORE DATA (1=YES, 0=NO)? 0
```

```
1 REM - OPTION 110,150
10 PRINT "□DAYS BETWEEN TWO DATES"
 .
 .
 .
90 Y=Y1
100 GOSUB 350
109 REM - SAVE COMPUTED NUMBER OF DAYS IN N
 .
 .
 .
140 Y=Y2
150 GOSUB 350
159 REM - CALCULATE DIFFERENCE AND PRINT
 .
 .
 .
227 REM - SUBROUTINE TO COMPUTE NUMBER OF DAYS FROM 0/0/0/ TO M/D/Y
 .
 .
 .
349 REM - TABLE OF NUMBER OF DAYS FROM FIRST OF YEAR TO FIRST OF EACH
   MONTH
 .
 .
 .
460 END
```

Anglo to Metric

This program converts a measure given in anglo units to metric units. The conversions available in this program are as follows:

1 Inches to centimeters
2 Feet to centimeters
3 Feet to meters
4 Yards to meters
5 Miles to kilometers
6 Teaspoons to cubic centimeters
7 Tablespoons to cubic centimeters
8 Cups to liters
9 Pints to liters
10 Quarts to liters
11 Gallons to liters
12 Bushels to liters
13 Pecks to liters
14 Ounces to grams
15 Pounds to kilograms
16 Tons to kilograms
17 Degrees Fahrenheit to degrees Celsius

You must provide the value of the anglo measurement and the number of the conversion (1 - 17 as listed above) which you wish to perform.

Example:

Perform the following conversions:

8.5 miles to kilometers
75° Fahrenheit to degrees Celsius
10 gallons to liters

```
ANGLO TO METRIC

(TO END PROGRAM TYPE 0)
WHICH CONVERSION DO YOU NEED? 5

VALUE TO BE CONVERTED? 8.5

 8.5 MILES = 13.6765 KILOMETERS


WHICH CONVERSION DO YOU NEED? 17

VALUE TO BE CONVERTED? 75

 75 DEGREES FAHRENHEIT = 23.8888889 CELSIUS
```

```
WHICH CONVERSION DO YOU NEED? 11

VALUE TO BE CONVERTED? 10

 10 GALLONS = 37.85 LITERS


WHICH CONVERSION DO YOU NEED? 0
```

```
10 PRINT "□ANGLO TO METRIC"
20 PRINT
29 REM - ESTABLISH VARIABLES FOR 17 CONVERSION FACTORS
30 DIM C(17)
39 REM - LOOP TO ASSIGN CONVERSION FACTORS INTO C()
40 FOR N=1 TO 17
50 READ C(N)
60 NEXT N
69 REM - DATA TABLE OF SEVENTEEN CONVERSION FACTORS
70 DATA 2.540,30.480,.3048,.9144
71 DATA 1.609,4.929,14.788,.2366
72 DATA .4732,.9463,3.785,35.24
73 DATA 8.809,28.3495,.4536,907.2
80 DATA .6214
89 REM - GET NUMBER OF CONVERSON FROM PROGRAM DESCRIPTION
90 PRINT "(TO END PROGRAM TYPE 0)"
100 PRINT "WHICH CONVERSION ";
101 PRINT "DO YOU NEED";
110 INPUT N:PRINT
119 REM - END PROGRAM?
120 IF N=0 THEN 540
129 REM - CONVERSIN AVAILABLE?
130 IF N>17 THEN 100
140 PRINT "VALUE TO BE CONVERTED";
150 INPUT I:PRINT
159 REM - PERFORM CONVERSION USING PROPER CONVERSION FACTOR
160 R=I*C(N)
169 REM - DIRECT PROGRAM TO PROPER CONVERSION UNITS, PRINT RESULTS
170 IF N<10 THEN 175
173 ON N-9  GOTO 360,380,400,420,440,460,480,500
175 ON N GOTO 180,200,220,240,260,280,300,320,340
180 PRINT I;"INCHES =";R;"CENTIMETERS"
190 GOTO 520
200 PRINT I;"FEET =";R;"CENTIMETERS"
210 GOTO 520
220 PRINT I;"FEET =";R;"METERS"
240 PRINT I;"YARDS =";R;"METERS"
250 GOTO 520
260 PRINT I;"MILES =";R;"KILOMETERS"
270 GOTO 520
280 PRINT I;"TSP. =";R;
281 PRINT "CUBIC CENTIMETERS"
290 GOTO 520
300 PRINT I;"TBSP. =";R;
301 PRINT "CUBIC CENTIMETERS"
310 GOTO 520
320 PRINT I;"CUPS =";R;"LITERS"
```

```
330 GOTO 520
340 PRINT I;"PINTS =";R;"LITERS"
350 GOTO 520
360 PRINT I;"QUARTS =";R;"LITERS"
370 GOTO 520
380 PRINT I;"GALLONS =";R;"LITERS"
390 GOTO 520
400 PRINT I;"BUSHELS =";R;"LITERS"
410 GOTO 520
420 PRINT I;"PECKS =";R;"LITERS"
430 GOTO 520
440 PRINT I;"OUNCES =";R;"GRAMS"
450 GOTO 520
460 PRINT I;"POUNDS =";R;"KILOGRAMS"
470 GOTO 520
480 PRINT I;"TONS =";R;"KILOGRAMS"
490 GOTO 520
499 REM - CONVERT FROM DEGREES FARENHEIT TO CELSIUS
500 R=(I-32)*5/9
510 PRINT I;"DEGREES FAHRENHEIT =";R;
511 PRINT "CELSIUS"
520 PRINT:PRINT
529 REM - RESTART PROGRAM
530 GOTO 100
540 END
```

Alphabetize

This program alphabetizes a list of words or phrases.

Numbers may be part of an alphanumeric phrase. However, they will not be put into numeric order unless they contain the same number of digits. Numbers with fewer digits must be justified to the right by prefixing zeros. Thus, if the numbers you are sorting range into the hundreds, the number 13 would be entered as 013.

To save memory space, the array at statement 70 should be limited to the maximum number of terms you wish alphabetized. The dimension statement should be altered in the following manner:

70 DIM A$(*N*)

where *N* = the number of items to be alphabetized.

Example:

Alphabetize the following names:

Robert Wilson
Susan W. James
Kent Smith
Michael Mitchell
Ann T. McGowan
Alexander Lee II
Mary Mitchell
David Bowers
Steven Evans
Carol Jameson
Linda North

```
70 DIM A$(11)
RUN


ALPHABETIZE

(TO END PROGRAM ENTER 0)
NUMBER OF ITEMS? 11
ITEM 1 ? WILSON ROBERT
ITEM 2 ? JAMES SUSAN W.
ITEM 3 ? SMITH KENT
ITEM 4 ? MITCHELL MICHAEL
ITEM 5 ? MCGOWAN ANN T.
ITEM 6 ? LEE ANEXANDER II
ITEM 7 ? MITCHELL MARY
ITEM 8 ? BOWERS DAVID
ITEM 9 ? EVANS STEVEN
ITEM 10 ? JAMESON CAROL
ITEM 11 ? NORTH LINDA
BOWERS DAVID
EVANS STEVEN
JAMES SUSAN W.
```

```
JAMESON CAROL
LEE ALEXANDER II
MCGOWAN ANN T.
MITCHELL MARY
MITCHELL MICHAEL
NORTH LINDA
SMITH KENT
WILSON ROBERT

NUMBER OR ITEMS? 0
```

```
10 PRINT "□ALPHABETIZE"
20 PRINT
30 PRINT "(TO END PROGRAM ENTER 0)"
40 PRINT "NUMBER OF ITEMS";
50 INPUT N
60 IF N=0 THEN 330
69 REM - LIMIT ARRAY TO MAXIMUM NUMBER OF ITEMS TO BE ENTERED IN ONE RUN
70 DIM A$(25)
80 FOR I=1 TO N
90 PRINT "ITEM";I;
100 INPUT A$(I)
110 NEXT I
120 M=N
127 REM - THE SORT TECHNIQUE USED COMPARES DATA ITEMS IN DIMINISHING
   INCREMENTS
128 REM - THE FIRST PASS COMPARES ITEMS N/2 ELEMENTS APART, THE SECOND
   (N/2)/2
129 REM - ELEMENTS APART, AND SO ON UNTIL THE INCREMENT IS EXHAUSTED.
130 T=M/2
131 M=INT(T)
140 IF M=0 THEN 280
150 K=N-M
160 J=1
170 I=J
180 L=I+M
190 IF A$(I)<=A$(L)THEN 250
200 T$=A$(I)
210 A$(I)=A$(L)
220 A$(L)=T$
230 I=I-M
240 IF I>=1 THEN 180
250 J=J+1
260 IF J>K THEN 130
270 GOTO 170
280 FOR I=1 TO N
281 REM - CHECK FOR FULL SCREEN (20 LINES)
282 IF I/20<>INT(I/20) THEN 290
283 REM - WAIT FOR OPERATOR CUE TO GO TO NEXT SCREEN
284 PRINT "ENTER 'C' TO CONTINUE";
285 INPUT W$
290 PRINT A$(I)
300 NEXT I
310 PRINT
319 REM - RESTART PROGRAM
320 GOTO 40
330 END
```

OPTION

You may wish your list alphabetized in reverse, or from highest to lowest. The program changes necessary are listed following the example below.

Example:

The scores on a math test range from 82 to 117. Put the students in order according to their scores, from highest to lowest.

```
 89 Bowers
102 Evans
111 James
100 Jameson
 99 Lee
117 McGowan
102 Mitchell
 82 Mitchell
 97 North
 91 Smith
108 Wilson
```

```
70 DIM A$(11)
RUN


ALPHABETIZE

(TO END PROGRAM ENTER 0)
NUMBER OF ITEMS? 11
ITEM 1 ? 089 BOWERS
ITEM 2 ? 102 EVANS
ITEM 3 ? 111 JAMES
ITEM 4 ? 100 JAMESON
ITEM 5 ? 099 LEE
ITEM 6 ? 117 MCGOWAN
ITEM 7 ? 102 MITCHELL
ITEM 8 ? 082 MITCHELL
ITEM 9 ? 097 NORTH
ITEM 10 ? 091 SMITH
ITEM 11 ? 108 WILSON
117 MCGOWAN
111 JAMES
108 WILSON
102 MITCHELL
102 EVANS
100 JAMESON
099 LEE
097 NORTH
091 SMITH
089 BOWERS
082 MITCHELL

NUMBER OF ITEMS? 0
```

```
1 REM - ALPHABETIZE -OPTION 190
4 REM - LIMIT ARRAY TO MAXIMUM NUMBER OF ITEMS TO BE ENTERED IN ONE RUN
5 DIM A$(25)
7 PRINT "□"
10 PRINT "ALPHABETIZE"
 .
 .
 .
180 L=I+M
190 IF A$(I)>=A$(L) THEN 250
200 T$=A$(I)
 .
 .
 .
330 END
```

References

Mendenhall, William, et al., *Statistics: A Tool for the Social Sciences.* North Scituate, Massachusetts: Duxbury Press, 1974.

Paige, Lowell J. and J. Dean Swift, *Elements of Linear Algebra.* Boston: Ginn and Company, 1961.

Sakarovitch, M., *Notes on Linear Programming.* New York: Van Nostrand Reinhold Company, 1971.

Spiegel, Murray R., *Theory and Problems of Statistics.* New York: Schaum's Outline Series, Schaum Publishing Company, 1961.

Thomas, George B., Jr., *Calculus and Analytic Geometry,* part one, 4th ed. Reading Massachusetts: Addison-Wesley Publishing Company, 1968.

U.S. Department of Commerce, *Handbook of Mathematical Functions.* National Bureau of Standards, Applied Mathematics Series 55, 1964.